

## REPORT DOCUMENTATION PAGE

AFRL-SR-BL-TR-98-

0155

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Service, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Project, Washington, DC 20503.

Completing and reviewing  
this report for information

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE Jan 1998		3. REPORT TYPE AND DATES COVERED FINAL TECH REPORT, 01 OCT 92 to 30 SEP 97	
4. TITLE AND SUBTITLE Formulation and Analysis of Stable Time-Stepping Algorithms for Contact Problems				5. FUNDING NUMBERS F49620-92-J-0543	
6. AUTHOR(S) Professor Eva G. Petocz					
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Div. of Mechanics & Computation, Dept of Mechanical Engineering Stanford University Stanford CA 94305				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) AFOSR/NM 110 Duncan Avenue Suite B115 Bolling AFB DC 20332-8050				10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES					
12a. DISTRIBUTION AVAILABILITY STATEMENT Approved for public release; distribution unlimited.					
13. ABSTRACT (Maximum 200 words) The formulation of stable time-stepping algorithms for dynamic contact problems, both frictionless and frictional, is presented. Special attention is given to the properties of the underlying continuum problem to serve as guidelines for the development of the algorithms. The proposed method conserves linear and angular momenta, and, in the frictionless case, conserves the energy by means of a restoration potential. Coulomb's friction law is used to model the friction phenomenon; the scheme presented herein is unconditionally dissipative, just as the physical system is. The scheme has been enhanced by the enforcement of a constraint on the velocities, in addition to the unilateral (impenetrability) constraint imposed on the displacements; this enhancement does not disturb the conservation/restoration properties. Numerical dissipation may also be added to stabilize the scheme for problems with high frequency energy modes. A multibody implementation is presented to show the versatility of the algorithm. In this implementation, the contact detection scheme includes an efficient sorting procedure which makes large scale simulations possible. Lastly, various numerical examples show the stability and robustness of the scheme.					
14. SUBJECT TERMS Keywords: algorithm, robustness, friction phenomenon				15. NUMBER OF PAGES 132	
				16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL		

19980205 087

**FORMULATION AND ANALYSIS OF  
STABLE TIME-STEPPING ALGORITHMS  
FOR CONTACT PROBLEMS**

By Eva G. Petőcz

Research supported by the AFOSR under contract  
No. 2-DJA-826 with Stanford University  
and F49620-97-1-0196 with UC Berkeley.

*F49620-92-J-0543*

SUDMC Report No. 98-02  
January 1998

DIVISION  
OF  
**MECHANICS  
AND  
COMPUTATION**

DEPARTMENT  
OF  
**MECHANICAL  
ENGINEERING**



**STANFORD  
UNIVERSITY**  
STANFORD  
CALIFORNIA  
94305

© Copyright 1997 by Eva G. Petőcz

All Rights Reserved

**DTIC QUALITY INSPECTED 3**

# Abstract

The formulation of stable time-stepping algorithms for dynamic contact problems, both frictionless and frictional, is presented. Special attention is given to the properties of the underlying continuum problem to serve as guidelines for the development of the algorithms. The proposed method conserves linear and angular momenta, and, in the frictionless case, conserves the energy by means of a restoration potential. Coulomb's friction law is used to model the friction phenomenon; the scheme presented herein is unconditionally dissipative, just as the physical system is.

The scheme has been enhanced by the enforcement of a constraint on the velocities, in addition to the unilateral (impenetrability) constraint imposed on the displacements; this enhancement does not disturb the conservation/restoration properties. Numerical dissipation may also be added to stabilize the scheme for problems with high frequency energy modes.

A multibody implementation is presented to show the versatility of the algorithm. In this implementation, the contact detection scheme includes an efficient sorting procedure which makes large scale simulations possible.

Lastly, various numerical examples show the stability and robustness of the scheme.



# Acknowledgements

I wish to acknowledge Prof. Juan C. Simo for giving me the opportunity to come to Stanford to continue my education. It is to the memory of him as an exceptional teacher and mentor that I dedicate this dissertation.

I am greatly indebted to Prof. Francisco Armero, my advisor, for his patient instruction and guidance throughout this challenging pursuit. I am grateful to Prof. Tom Hughes and Prof. Robert Taylor for their constant support and encouragement.

This work could not have been done without the help of two good friends and computer experts, Michael Grant and Patrick Klein. I thank Chandler Fulton for his careful proofreading of this dissertation.

Financial support for this research has been provided by the AFOSR under contract no. 2-DJA-826 with Stanford University and no. F49620-97-1-0196 with the University of California at Berkeley. This support is gratefully acknowledged.

I also want to thank my family and all my friends.

# Contents

Abstract	iv
Acknowledgements	v
List of Tables	xi
List of Figures	xii
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Background . . . . .	2
1.3 Goals . . . . .	3
1.4 Outline . . . . .	5
<b>2 Problem Definition</b>	<b>6</b>
2.1 Introduction . . . . .	6
2.2 Notation . . . . .	7
2.3 Local governing equations . . . . .	10
2.3.1 Contact definition . . . . .	11

2.3.1.1	Closest point projection . . . . .	11
2.3.2	Contact constraint . . . . .	12
2.3.3	Persistent contact . . . . .	14
2.4	The frictional problem . . . . .	16
2.4.1	Contact kinematics . . . . .	16
2.4.2	Coulomb friction . . . . .	19
<b>3</b>	<b>The Weak Formulation</b>	<b>22</b>
3.1	Introduction . . . . .	22
3.2	Weak form of the governing equations . . . . .	22
3.3	Properties of the weak formulation . . . . .	24
3.3.1	Conservation properties of contact . . . . .	25
3.3.1.1	Conservation of linear momentum . . . . .	25
3.3.1.2	Conservation of angular momentum . . . . .	26
3.3.1.3	Evolution of the energy . . . . .	27
<b>4</b>	<b>Finite Element Implementation</b>	<b>30</b>
4.1	Introduction . . . . .	30
4.2	Time discretization of the weak equation . . . . .	30
4.3	Conserving properties of the algorithm . . . . .	32
4.3.1	Internal linear momentum . . . . .	32
4.3.2	Internal angular momentum . . . . .	33
4.3.3	Internal energy . . . . .	35
4.4	The semi-discrete equations . . . . .	36

4.5	Space discretization . . . . .	37
4.5.1	Closest point projection in the space-discretized setting . . . . .	38
4.6	Temporal and space discretization . . . . .	41
<b>5</b>	<b>Energy Restoring Momentum Conserving Algorithm for Frictionless Dynamic Contact</b>	<b>43</b>
5.1	Introduction . . . . .	43
5.2	Energy restoring/momentum conserving scheme . . . . .	44
5.2.1	Time discretization of the weak form . . . . .	44
5.2.2	Energy restoration . . . . .	48
5.2.3	Space discretization . . . . .	49
5.2.4	Contact/release conditions . . . . .	50
5.3	Enforcement of the gap velocity constraint . . . . .	50
5.3.1	Properties of the proposed scheme . . . . .	53
5.3.1.1	Energy restoration . . . . .	56
5.3.2	Finite element implementation . . . . .	56
5.4	Addition of energy dissipation . . . . .	57
<b>6</b>	<b>Frictional Dynamic Contact</b>	<b>60</b>
6.1	Introduction . . . . .	60
6.2	Dissipative friction algorithm . . . . .	60
6.2.1	Split operator integration algorithm . . . . .	62
6.3	Energy evolution of the frictional algorithm . . . . .	65
6.4	Space discretization . . . . .	68

<b>7</b>	<b>Sorting Algorithm</b>	<b>70</b>
7.1	Introduction . . . . .	70
7.2	Overview of various sorting techniques . . . . .	71
7.3	Binary space partitioning . . . . .	74
7.3.1	Binary tree structure . . . . .	74
7.3.2	Space partitioning using a binary tree . . . . .	75
7.3.2.1	Construction of the binary tree example . . . . .	76
7.3.2.2	Binary tree construction program . . . . .	77
7.4	Sorting using binary tree . . . . .	78
7.4.1	Traversing the binary tree . . . . .	79
7.4.1.1	Preorder scheme . . . . .	79
7.5	Implementation of a sorting algorithm . . . . .	80
7.5.1	Binary tree . . . . .	81
7.5.2	Body, surface and particle . . . . .	81
7.5.3	Linked lists . . . . .	82
7.5.4	Auxiliary tools . . . . .	84
7.5.5	Basic algorithm . . . . .	84
7.6	Computation times . . . . .	86
<b>8</b>	<b>Numerical Examples</b>	<b>90</b>
8.1	Introduction . . . . .	90
8.2	Frictionless contact simulations . . . . .	90
8.2.1	Impact of a rod on a rigid wall . . . . .	90

8.2.2	Impact of two nonlinear elastic cylinders . . . . .	98
8.2.2.1	Quasi-rigid cylinders . . . . .	98
8.2.2.2	Soft cylinders . . . . .	101
8.3	Frictional contact simulations . . . . .	104
8.3.1	Impact of a cylinder against a rigid wall . . . . .	104
8.3.2	Forging of an elastic block against a rigid foundation . . . . .	107
8.3.3	Oblique impact of two infinite blocks . . . . .	108
8.4	Multibody contact . . . . .	112
8.4.1	Impact of 9 elastic disks . . . . .	112
8.4.2	Impact of 49 quasi-rigid disks . . . . .	115
<b>9</b>	<b>Conclusions</b>	<b>118</b>
9.1	Closure . . . . .	118
9.2	Future work . . . . .	120
<b>A</b>	<b>Consistent Linearization of the Proposed Schemes</b>	<b>121</b>
A.1	The linearized equations . . . . .	121
A.2	The contact stiffness . . . . .	123
A.2.1	The normal contact stiffness . . . . .	124
A.2.2	The tangent contact stiffness . . . . .	126

# List of Tables

5.1	Contact/release logic. . . . .	51
7.1	Basic algorithm for the sort/search procedure . . . . .	85
7.2	Computing times in seconds for various numbers of bodies. . . . .	87
7.3	Computing times in seconds for various numbers of bodies. . . . .	89

# List of Figures

2.1	Schematic drawing of the motion of two bodies . . . . .	8
2.2	Geometric interpretation of the gap function $g$ . . . . .	13
2.3	Parametrization of the contact surfaces $\Gamma^2$ and $\gamma^2$ . . . . .	16
4.1	Schematic drawing of the closest point procedure on a discrete space setting in two dimensions. In the case shown, contact is detected and the master surface element is defined by 4 nodes. . . . .	39
4.2	Schematic drawing of two contact situations involving with two dimensional linear surface elements. Slave nodes are denoted by $S$ and master nodes by $M$ . . . . .	41
7.1	Schematic drawing of a binary tree . . . . .	74
7.2	Schematic drawing of a set of particles within a square working space . . . . .	76
7.3	Binary tree structure corresponding to the example shown in Figure 7.2 . . . . .	77
7.4	Schematic drawing of a body and its corresponding buffer zone . . . . .	79
7.5	Contact detection CPU time for the algorithm with sorting procedure. Computational data ( o ); regression analysis (—). . . . .	87
7.6	Contact detection CPU time for the algorithm with no sorting procedure. Computational data ( o ); regression analysis (—). . . . .	88
8.1	Impact of an elastic rod against a rigid wall. Schematic drawing of the problem. . . . .	91



8.2	Impact of an elastic rod against a rigid wall. Plots comparing <b>Algorithm 1</b> ( $\cdots$ ) and <b>Algorithm 4</b> ( $-$ ). . . . .	94
8.3	Impact of an elastic rod against a rigid wall. Plots comparing <b>Algorithm 2</b> ( $\cdots$ ) and <b>Algorithm 4</b> ( $-$ ). . . . .	95
8.4	Impact of an elastic rod against a rigid wall. Plots comparing <b>Algorithm 3</b> ( $\cdots$ ) and <b>Algorithm 4</b> ( $-$ ). . . . .	96
8.5	Impact of an elastic rod against a rigid wall. Plots comparing <b>Algorithm 5</b> with $\theta = 0.9$ ( $\cdots$ ) and <b>Algorithm 4</b> ( $-$ ). . . . .	97
8.6	Impact of quasi-rigid cylinders. Evolution of the system . . . . .	99
8.7	Impact of quasi-rigid cylinders. Evolution of the linear momenta, angular momentum and the energy of the system. Plots comparing <b>Algorithm 2</b> ( $\cdots$ ) and the proposed scheme ( $-$ ). . . . .	100
8.8	Impact between two soft elastic cylinders. Evolution of the dynamical system. . . . .	102
8.9	Impact between two soft elastic cylinders. Energy evolution of the dynamical system. Plots comparing <b>Algorithm 2</b> ( $\cdots$ ) and <b>Algorithm 4</b> ( $-$ ). . . . .	103
8.10	Impact of an elastic cylinder against a rigid wall. Three instances of the evolution of the frictionless case, at times $t = 0, 6, 12$ . . . . .	105
8.11	Impact of an elastic cylinder against a rigid wall. Three instances of the evolution of the frictional case ( $\mu = 0.2$ ), at times $t = 0, 6, 12$ . . . . .	105
8.12	Impact of an elastic cylinder against a rigid wall. Energy evolution of the frictionless case. Plots comparing the proposed scheme ( $-$ ) and the mid-point rule ( $\cdots$ ). . . . .	106
8.13	Impact of an elastic cylinder against a rigid wall. Energy evolution for the frictional case ( $\mu = 0.2$ ). Plots comparing the proposed scheme ( $-$ ) and the mid-point rule ( $\cdots$ ). . . . .	106
8.14	Forging of an elastic block against a rigid foundation. Schematic drawing of the problem. . . . .	107

8.15 Forging of an elastic block against a rigid foundation. Deformed mesh. . . .	108
8.16 Forging of an elastic block against a rigid foundation. Stress curves along the base of the block. . . . .	109
8.17 Oblique impact of two elastic blocks. Schematic drawing of the problem. . .	109
8.18 Oblique impact of two elastic blocks. Displacement of point A. . . . .	110
8.19 Oblique impact of two elastic blocks. Deformed mesh at $t = 0.1$ . . . . .	111
8.20 Oblique impact of two elastic blocks. Stress $\sigma_{xx}$ at $t = 0.1$ for the frictional case. . . . .	111
8.21 Oblique impact of two elastic blocks. Stress $\sigma_{yy}$ at $t = 0.1$ for the frictional case. . . . .	112
8.22 Impact of 9 elastic disks. Evolution of the system. . . . .	113
8.23 Impact of 9 elastic disks. Evolution of the energy of the system. Plots com- paring <b>Algorithm 2</b> ( $\cdots$ ) and the proposed scheme( $—$ ). . . . .	114
8.24 Impact of 49 quasi-rigid disks. Evolution of the system, . . . . .	116
8.25 Impact of 49 quasi-rigid disks. Evolution of the system (continued). . . . .	117

8.2	Impact of an elastic rod against a rigid wall. Plots comparing <b>Algorithm 1</b> ( $\cdots$ ) and <b>Algorithm 4</b> ( $—$ ).	94
8.3	Impact of an elastic rod against a rigid wall. Plots comparing <b>Algorithm 2</b> ( $\cdots$ ) and <b>Algorithm 4</b> ( $—$ ).	95
8.4	Impact of an elastic rod against a rigid wall. Plots comparing <b>Algorithm 3</b> ( $\cdots$ ) and <b>Algorithm 4</b> ( $—$ ).	96
8.5	Impact of an elastic rod against a rigid wall. Plots comparing <b>Algorithm 5</b> with $\theta = 0.9$ ( $\cdots$ ) and <b>Algorithm 4</b> ( $—$ ).	97
8.6	Impact of quasi-rigid cylinders. Evolution of the system	99
8.7	Impact of quasi-rigid cylinders. Evolution of the linear momenta, angular momentum and the energy of the system. Plots comparing <b>Algorithm 2</b> ( $\cdots$ ) and the proposed scheme ( $—$ ).	100
8.8	Impact between two soft elastic cylinders. Evolution of the dynamical system.	102
8.9	Impact between two soft elastic cylinders. Energy evolution of the dynamical system. Plots comparing <b>Algorithm 2</b> ( $\cdots$ ) and <b>Algorithm 4</b> ( $—$ ).	103
8.10	Impact of an elastic cylinder against a rigid wall. Three instances of the evolution of the frictionless case, at times $t = 0, 6, 12$ .	105
8.11	Impact of an elastic cylinder against a rigid wall. Three instances of the evolution of the frictional case ( $\mu = 0.2$ ), at times $t = 0, 6, 12$ .	105
8.12	Impact of an elastic cylinder against a rigid wall. Energy evolution of the frictionless case. Plots comparing the proposed scheme ( $—$ ) and the mid-point rule ( $\cdots$ ).	106
8.13	Impact of an elastic cylinder against a rigid wall. Energy evolution for the frictional case ( $\mu = 0.2$ ). Plots comparing the proposed scheme ( $—$ ) and the mid-point rule ( $\cdots$ ).	106
8.14	Forging of an elastic block against a rigid foundation. Schematic drawing of the problem.	107

# Chapter 1

## Introduction

### 1.1 Motivation

Contact or impact situations are present in many engineering applications, such as metal forming processes and crash-worthiness testing. The modeling of contact problems is a highly nonlinear situation due to the fact that it is a *unilaterally constrained* problem. The introduction of friction between the contacting bodies is another important source of nonlinearity. The frictionless case is in fact a unilaterally constrained Hamiltonian system giving rise to many conserving properties. The challenge is for the algorithmic schemes to simulate these conservation properties accurately.

To circumvent the difficulties associated with multiple nonlinearities, many explicit schemes have been developed in the past. But one of the main drawbacks of this type of scheme becomes evident when modeling dynamic problems. Explicit schemes have limited stability properties, sometimes leading to poor enforcement of the constraints. On the other hand, some implicit schemes are stable in a linear regime, but may lose their stability properties in a nonlinear problem, giving rise to a non-physical energy increase in the system. Characteristic examples are the trapezoidal and mid-point rules, which are energy conserving schemes in the linear regime but show significant increase in the energy of the system for nonlinear problems, and may even result in numerical blow-ups.

In addition to the impenetrability constraint, the system must be constrained so that the rate of separation during contact is zero. In fact, higher order time derivatives of this measure of separation are also required to vanish. A poor enforcement of these constraints yields oscillations between contact and release states which damage the robustness of a contact scheme.

The source of the oscillatory behaviour is the inability of a space and time discretized formulation to represent the shock wave that reverses the velocity of the contact point at the instant of contact. The ineffectiveness of representing these high frequency energy modes, known also as the Gibbs phenomenon, is a well known source of difficulty in impact algorithms where the short term behaviour is of interest.

The simulation of contact problems also involves contact detection algorithms which may affect the overall efficiency of the scheme, especially in cases of many body problems. A robust contact scheme should be able to resolve multiple collisions among multiple bodies. All of these considerations play an important role in the design of both the contact detection algorithm and the overall contact formulation.

## 1.2 Background

Since the development of the finite element method in the late 1950's, numerical solutions to contact problems have been investigated intensively by many researchers. Earlier work done by FRANCAVILLA & ZIENKIEWICZ [10] on a flexibility approach and by HUGHES ET AL[15] on the use of Lagrange multiplier methods, contributed to the development of robust finite element methods. A fairly comprehensive overview of the numerical methodologies used to solve the quasi-static contact problem can be found in ZHONG & MACKERLE[38].

Descriptions of finite elements methods for dynamic contact problems may be found in BELYTSCHKO & NEAL[5] who developed *pinball* methodologies; in CARPENTER ET AL[7], who worked on the development of Lagrange multipliers for the enforcement of unilateral constraints; and in HALLQUIST ET AL[12], who developed the concept of master and slave

methodology. All of these formulations involve explicit integrators.

A comprehensive review of both the frictionless and frictional dynamic contact problems is presented in KIKUCHI & ODEN[16].

Enforcement of higher order constraints (i.e. vanishing of rate of separation) in addition to the impenetrability constraint has been researched by LEE[21] and TAYLOR & PAPADOPOULOS[33].

When dealing with nonlinear dynamical systems, stability issues become critical in the simulation of dynamic contact problems. In the area of elastodynamics, the consideration of energy/momentum conserving algorithms leading to stable schemes, is described in SIMO & TARNOW [30] and SIMO ET AL[31]. Examples of the work concentrating on stability and conservation properties include MUNJIZA ET AL[23] and ARMERO & PETŐCZ[1].

In the area of frictional contact problems, formulations using both penalty regularization and augmented Lagrangians has been published by LAURSEN & SIMO[19], ODEN & MARTINS [25] and WRIGGERS ET AL[37].

Contact detection algorithms for multibody problems have been developed primarily by those in the discrete element method research community, who deal predominantly with rigid bodies. However, many ideas can be extracted from their work to be used in conjunction with the finite element method. A state-of-the-art review is included in WILLIAMS & O'CONNOR [35]. Some examples of contact detection algorithms using tree data structures are presented in MUNJIZA ET AL[24] and BONET & PERAIRE[6].

## 1.3 Goals

The main goal of this work is to develop stable implicit time-stepping algorithms for dynamic contact problems. We have analyzed the physical properties of the continuum dynamical system and developed our algorithm following these guidelines.

In a general contact problem among elastic bodies, in the absence of external forces and

imposed displacements, linear and angular momenta are conserved. The proposed scheme inherits these conservation properties by construction. Furthermore, in the frictionless case, the total energy of the system is also conserved. The algorithm presented in this work accomplishes the restoration of the total energy after the release of the contacting bodies. That is, the penalty regularization potential used to enforce the unilateral constraint is also used to restore the energy that was taken away from the system during contact. The energy restoration scheme is second order accurate and is unconditionally (energy) stable without relying on artificial numerical dissipation.

During persistent contact, in which two bodies may remain in contact for long periods of time, we have additional constraints which force the rate of separation between the contact surfaces to be zero.

As mentioned above, the presence of high frequency modes in impact problems tests the robustness of many contact schemes. Numerical dissipation is added to the developed scheme in order to circumvent the difficulties inherent in this problem. The amount of dissipation introduced into the scheme can be regulated to satisfy the requirements of each particular problem.

Friction phenomena are purely dissipative physical phenomena; hence our goal is to develop a frictional scheme which can guarantee positive energy dissipation under any circumstances. The proposed scheme is unconditionally dissipative.

The contact algorithm has to be easy to implement within a multibody contact formulation. The contact detection scheme is modified by means of object-oriented programming and intelligent database structures, to accommodate the new contact detection schemes.

By a series of numerical examples, we show the different properties of the contact scheme. One dimensional benchmark problems and two dimensional quasi static and dynamical examples clearly show the superior stability and conservation properties the proposed scheme over conventional schemes.

## 1.4 Outline

In Chapter 2 we shall describe the problem of contact between elastic bodies and explain the notation used in this work. We state the governing equations for both the frictionless and the frictional case. In Chapter 3, we develop the weak formulation for the previously stated strong form of the equations and study its properties. In Chapter 4, we describe the finite element implementation and the time-stepping procedure to solve the dynamical problem. Contact detection in a space discretized setting will also be discussed. We describe the algorithms developed for the frictionless contact problem and for the frictional case in Chapters 5 and 6, respectively, and we study their stability properties. In Chapter 7, we go on to describe the multibody implementation of the contact schemes developed in this work. In Chapter 8, we present various numerical examples in one and two dimensions to show the performance of our schemes. We draw conclusions and propose some future work in Chapter 9.



# Chapter 2

## Problem Definition

### 2.1 Introduction

In this chapter, we develop the equations that define the contact problem and the notation required to interpret them. The equations describe the continuum problem of contact between two or more bodies with large deformations, for both frictionless and frictional cases.

The equations are defined using the Lagrangian description, which is the most prevalent framework in computational solid mechanics. For the sake of generality, the equations and subsequent algorithmic solutions will be developed in a dynamic formulation but the work presented herein is also valid for the quasi-static case.

Contact is a unilaterally constrained problem in which we impose that two bodies may not penetrate each other. Relative tangent motion may be present and friction may be included in the statement of the problem. Many models have been proposed to describe the friction phenomena, however in the context of the finite element method most rely on the classical law of Coulomb. A description of other friction models may be found in ODEN & MARTINS [25].

We introduce basic notation in Section 2, describe the governing equations in their local form in Section 3 and describe the frictional problem in Section 4.

## 2.2 Notation

We describe the contact problem between two elastic bodies; special cases such as self-contact are excluded from the problem description. These restrictions are made only to simplify the notation and the formulation of the equations, but the scheme can be easily extended to more bodies. Details regarding a straightforward implementation to a system of multiple bodies is be developed in Chapter 7.

Figure 2.1, is a schematic drawing of two elastic bodies in their reference and deformed configurations. Each body is represented by an open set  $\Omega^{(i)}$  for  $i = 1, 2$ , and both reside in  $\mathbb{R}^{n_{dim}}$ , where  $n_{dim}$  is the number of space dimensions. These bodies undergo motions described at time  $t$  by the deformations  $\varphi^{(i)}(t)$  for  $i = 1, 2$ , where  $t$  belongs to the time interval  $\mathbb{I} = [0, T]$ . Thus, these motions are expressed via the following mappings:

$$\varphi^{(i)}(t) : \Omega^{(i)} \times \mathbb{I} \rightarrow \mathbb{R}^{n_{dim}} \quad \text{for } i = 1, 2. \quad (2.1)$$

For simplicity, we assume that the reference configurations  $\Omega^{(i)}$  for  $i = 1, 2$  are the initial configurations of the bodies at  $t = 0$ . We also assume that at time  $t = 0$ , there is no contact between the bodies; consequently, no contact forces are present.

We identify the material particles of each solid with the reference coordinate  $\mathbf{X} \in \cup_i \Omega^{(i)} \subset \mathbb{R}^{n_{dim}}$ . The current placement of the material particle  $\mathbf{X} \in \Omega^{(i)}$  at time  $t \in [0, T]$  is then expressed as  $\mathbf{x}^{(i)} := \varphi^{(i)}(\mathbf{X}, t)$ .

We denote the boundaries of each solid  $i$ , for  $i = 1, 2$ , by  $\Gamma^{(i)} := \partial\Omega^{(i)}$  in the reference configuration and by  $\gamma^{(i)} := \partial\varphi^{(i)}(\Omega^{(i)})$  in the current configuration. We denote by  $\gamma_c := \cap_i \gamma^{(i)}$  the common current boundary in contact between any two bodies (see Figure 2.1); analogously, in the reference configuration the common boundary at time  $t$  can be expressed by  $\Gamma_c := \varphi_t^{(i)-1}(\gamma_c)$ .

We denote by  $\mathbf{P}^{(i)}$  the nominal stress tensor (first Piola-Kirchhoff stress tensor) in each solid and we shall restrict our analysis to hyperelastic solids characterized by their respective

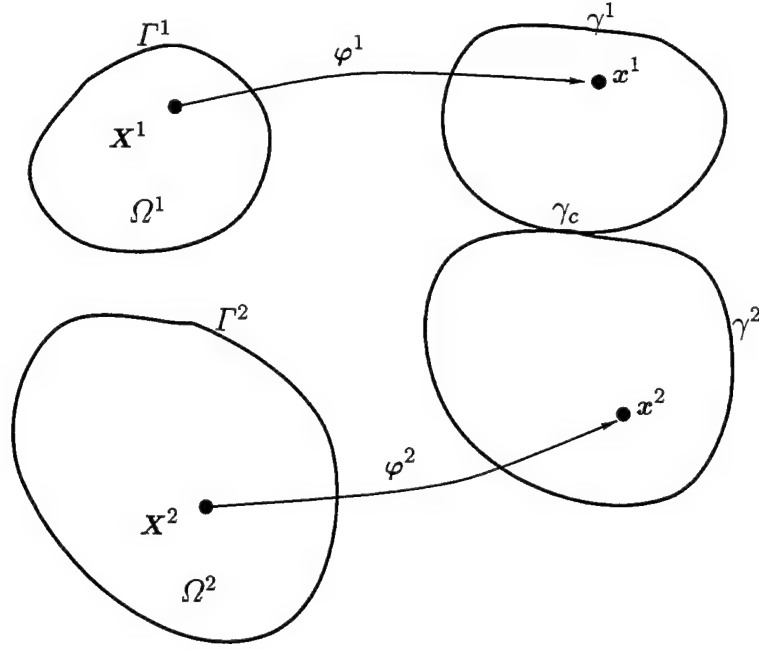


Figure 2.1: Schematic drawing of the motion of two bodies

stored energy functions  $W(\mathbf{F}^{(i)})$ , where  $\mathbf{F}^{(i)} = \text{Grad}\varphi^{(i)}$  is the deformation gradient of each solid. Thus, we have

$$\mathbf{P}^{(i)} = \frac{\partial W^{(i)}}{\partial \mathbf{F}^{(i)}}. \quad (2.2)$$

To satisfy the principle of material frame indifference, the stored energy function  $W^{(i)}$  is invariant under the action of the proper orthogonal group (the rotation group)  $SO(n_{dim})$ , that is,

$$W^{(i)}(Q\mathbf{F}^{(i)}) = W^{(i)}(\mathbf{F}^{(i)}) \quad \forall Q \in SO(n_{dim}). \quad (2.3)$$

Considering a one-parameter group of rotations  $Q(\eta)$  with

$$\left. \frac{d}{d\eta} Q(\eta) \right|_{\eta=0} = \hat{\mathbf{W}} \in so(n_{dim}), \quad (2.4)$$

where  $so(n_{dim})$  denotes a linear space of skew-symmetric tensors. Taking the derivative of (2.3) with respect to  $\eta$  and setting  $\eta = 0$ , we obtain the following relation:

$$\frac{\partial W^{(i)}}{\partial \mathbf{F}^{(i)}} \mathbf{F}^{(i)T} : \hat{\mathbf{W}} = 0 \quad \forall \hat{\mathbf{W}} \in so(n_{dim}), \quad (2.5)$$

implying the symmetry of the Kirchhoff stress tensor, that is:

$$\boldsymbol{\tau}^{(i)} := \frac{\partial W^{(i)}}{\partial \mathbf{F}^{(i)}} \mathbf{F}^{(i)T} = \boldsymbol{\tau}^{(i)T}. \quad (2.6)$$

The symmetry relation 2.6 leads to the classical conservation law of the total angular momentum as discussed below. Furthermore, a classical argument (see e.g. TRUESDELL & NOLL[34]) leads to the dependence of the stored energy function on the Green-Lagrange strain tensor defined as

$$\mathbf{E} = \frac{1}{2}(\mathbf{F}^T \mathbf{F} - \mathbf{1}). \quad (2.7)$$

The new expression for the stored energy function denoted by  $\bar{W}$  is then

$$W^{(i)}(\mathbf{F}^{(i)}) = \bar{W}^{(i)}(\mathbf{E}^{(i)}). \quad (2.8)$$

As an example, most of the simulations presented in this work use the Saint-Venant Kirchhoff model, characterized by:

$$W(\mathbf{F}) = \bar{W}(\mathbf{E}) = \frac{1}{2} \mathbf{E} : \mathbf{C} \mathbf{E}, \quad (2.9)$$

where  $\mathbf{C}$  denotes the material secant tangent.

We denote by  $\mathbf{v}^{(i)} := \dot{\boldsymbol{\varphi}}^{(i)}$  the material velocity field of the solid  $i$ , and  $\rho^{(i)}$  the corresponding reference density. The superimposed dot  $(\dot{\cdot})$  refers to the (material) derivative with respect to time  $t$ .

## 2.3 Local governing equations

In the absence of contact, the strong form of the local momentum balance, along with the boundary and initial conditions, is as follows:

For all  $t \in [0, T]$ ,  $\varphi^{(i)}$  ( $i = 1, 2$ ) must satisfy

$$\rho_0^{(i)} \frac{\partial^2}{\partial t^2} \varphi^{(i)} = \text{Div} \mathbf{P}^{(i)} + \mathbf{f}_{ext}^{(i)} \quad \text{in } \Omega^{(i)}, \quad (2.10)$$

$$\mathbf{P}^{(i)} \mathbf{n}_0^{(i)} = \bar{\mathbf{t}}^{(i)} \quad \text{on } \Gamma_\sigma^{(i)}, \quad (2.11)$$

$$\varphi^{(i)} = \bar{\varphi}^{(i)} \quad \text{on } \Gamma_\varphi^{(i)}, \quad (2.12)$$

and the initial conditions

$$\varphi^{(i)} \Big|_{t=0} = \mathbf{1} \quad (\text{the identity mapping}) \quad \text{in } \Omega^{(i)}, \quad (2.13)$$

$$\frac{\partial}{\partial t} \varphi^{(i)} \Big|_{t=0} = \mathbf{V}_0^{(i)} \quad \text{in } \Omega^{(i)}. \quad (2.14)$$

In the above equations,  $\rho_0^{(i)}$  is the reference density,  $\mathbf{n}_0^{(i)}$  is the outward normal in the reference configuration and  $\mathbf{P}^{(i)}$  represents the first Piola-Kirchhoff stress tensor for body  $i$ ; and  $\text{Div}$  represents the divergence operator in the reference geometry. Also, the mappings  $\mathbf{f}^{(i)} : \Omega^{(i)} \times \mathbb{I} \rightarrow \mathbb{R}^{n_{dim}}$ ,  $\bar{\mathbf{t}}_{ext}^{(i)} : \Gamma_\sigma^{(i)} \times \mathbb{I} \rightarrow \mathbb{R}^{n_{dim}}$ , and  $\bar{\varphi}^{(i)} : \Gamma_\varphi^{(i)} \times \mathbb{I} \rightarrow \mathbb{R}^{n_{dim}}$  represent the prescribed body forces, tractions and displacements, respectively, expressed in the reference configuration. Equation 2.13 is the mathematical expression which states that the initial configuration of the bodies coincides with the reference configuration. In equation 2.14,  $\mathbf{V}_0^{(i)}$  represents the prescribed initial velocity field.

For the above system of equations to describe a well-posed problem, we need to satisfy the following relations:

$$\Gamma^{(i)} \cup \Gamma_{\sigma}^{(i)} \cup \Gamma_{\varphi}^{(i)} = \partial\Omega^{(i)} \quad (2.15)$$

$$\Gamma^{(i)} \cap \Gamma_{\sigma}^{(i)} = \Gamma_{\sigma}^{(i)} \cap \Gamma_{\varphi}^{(i)} = \Gamma_{\varphi}^{(i)} \cap \Gamma^{(i)} = \emptyset. \quad (2.16)$$

A purely Neumann problem has no displacements imposed on the boundary of its bodies; thus  $\Gamma_{\varphi}^{(i)} = \emptyset$ . A purely Dirichlet problem has no tractions imposed on the boundary of its bodies; thus  $\Gamma_{\sigma}^{(i)} = \emptyset$ . In this dissertation, we work exclusively with Neumann type of problems as they have very distinct conservation properties which we want our algorithms to emulate. However, the schemes developed in this dissertation are not in any way restricted to this type of problems.

### 2.3.1 Contact definition

We now consider the addition of the contact constraints to the strong form presented in the previous section. To be able to do this we first need to define a measure of the closeness of two bodies. This is done by means of the gap function. The gap function measures the distance between one point on the surface of body 1 to a point on the surface of body 2. This mapping between a point of body 1 and its closest point in body 2 is called the *closest point projection* and its definition is detailed in the following section. The concept of closest point projection with its algorithmic interpretation has been extensively documented. Several examples can be seen in HALLQUIST ET AL[12] and WRIGGERS[36]. The definition of the closest point projection in the continuum setting is documented in LAURSEN & SIMO[19] and in SIMO & LAURSEN[29].

#### 2.3.1.1 Closest point projection

In what follows, we denote by  $\mathbf{X}$  a material point which belongs to the surface of body 1, i.e. to  $\Gamma^1$ . We define the gap function  $g(\mathbf{X})$  for a material point  $\mathbf{X} \in \Gamma^1$  as follows:

$$g(\mathbf{X}) := \boldsymbol{\nu} \cdot [\boldsymbol{\varphi}^1(\mathbf{X}) - \boldsymbol{\varphi}^2(\bar{\mathbf{Y}}(\mathbf{X}))] , \quad (2.17)$$

where the mapping  $\mathbf{Y} = \bar{\mathbf{Y}}(\mathbf{X}) \in \Gamma^2$  defines the closest point projection on the contact surface  $\Gamma^2$  at the current configuration of the solids, that is,

$$\bar{\mathbf{Y}}(\mathbf{X}) = \arg \min_{\mathbf{Y} \in \Gamma^2 \cap \Gamma_c} \{ \|\boldsymbol{\varphi}^1(\mathbf{X}) - \boldsymbol{\varphi}^2(\mathbf{Y})\| \} . \quad (2.18)$$

In the above equation,  $\|\cdot\|$  denotes the usual Euclidean vector norm and  $\boldsymbol{\nu} = \boldsymbol{\nu}(\bar{\mathbf{Y}}(\mathbf{X}))$  denotes the unit outward normal to the current contact boundary  $\gamma^2 \cap \gamma_c$ .

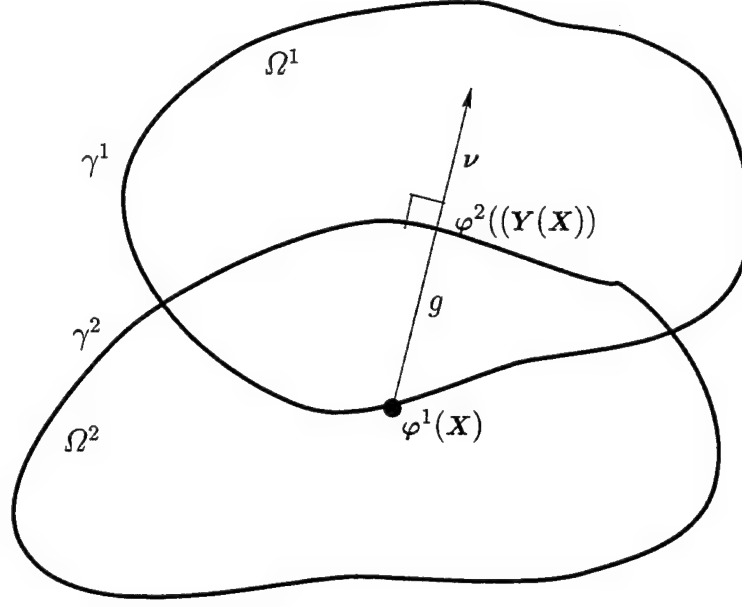
### 2.3.2 Contact constraint

The normal contact constraint enforces the physical condition of impenetrability and compressive interaction between bodies. For this purpose, we define one of the surfaces as the *slave surface or contactor surface* and the other as the *master surface or target surface*. We can interpret from the above that the master surface nodes (master nodes) define the surface of impenetrability for any node belonging to the slave surface (slave node).

For simplicity, we always choose the slave surface to be  $\Gamma^1$  and the master surface to be  $\Gamma^2$ . The impenetrability constraint is then established as

$$g(\mathbf{X}, t) \geq 0 \quad \forall \mathbf{X} \in \Gamma^1 . \quad (2.19)$$

The unilateral constraint is intimately related to the definition of the gap function  $g(\mathbf{X}, t)$  through the closest point projection. This definition is fairly straightforward for the continuum but gives rise to many options when we work with a space discretized formulation such as the finite element formulation. One of these options is developed in this work. Figure 2.2 depicts a violation of condition 2.19 and gives a physical interpretation of the definition of the gap function  $g(\mathbf{X}, t)$ .


 Figure 2.2: Geometric interpretation of the gap function  $g$ 

**Remark 2.1** In the continuum setting the distinction between master and slave surfaces breaks down, because if one point belonging to the slave surface lies on the other side of the master surface, the reverse is automatically true. But in a discretized space setting, one can find many cases where the penetration state would be altered if we interchanged the definitions of slave and master surface. We develop this idea in Chapter 4 when we explain the finite element implementation of the contact problem.

When two bodies come into contact, contact forces are generated along the common boundary  $\gamma_c$ . We recognize two contributions to this traction: a normal component  $t_N$  which originates from the impenetrability constraint and a tangent component  $t_T$  originating from the frictional phenomena. Thus, we have

$$\tilde{t}^{(i)} := t_N^{(i)} + t_T^{(i)} \quad (2.20)$$

$$= t_N^{(i)} \nu - t_{T\beta}^{(i)} \tau^\beta. \quad (2.21)$$

To enforce the principle of action and reaction across the the contact interface, so as to



conserve linear momentum, we require that the differential contact force induced on the surface of body 2 at  $\bar{Y}$  is equal and opposite to that produced on body 1 at  $\mathbf{X}$ . That is,

$$\tilde{\mathbf{t}}^1 d\Gamma^1 + \tilde{\mathbf{t}}^2 d\Gamma^2 = 0, \quad (2.22)$$

where  $\tilde{\mathbf{t}}^{(i)}$  is the traction vector due to contact at the common boundary  $\gamma_c$  belonging to the surface of each body  $i$ . In addition, we assume that no tensile (normal) tractions occur due to contact, so that

$$\tilde{\mathbf{t}}^{(i)} \cdot \boldsymbol{\nu}^{(i)} \geq 0, \quad (2.23)$$

where  $\boldsymbol{\nu}^{(i)}$  is the unit outward normal vector belonging to a point on the surface of body  $i$ . This condition ensures that the two surfaces will not become “glued” together once contact occurs. The inequality 2.23 can be rewritten as

$$t_N \geq 0. \quad (2.24)$$

### 2.3.3 Persistent contact

Persistent contact occurs when two surfaces remain in contact for a period of time, i.e. contact and release do not occur instantaneously. During the period of contact, then, all the higher time derivatives of the gap function  $g$  vanish (see e.g. TAYLOR & PAPADOPOULOS[33]). That is,

$$\frac{d^n g}{dt^n} = 0 \quad \text{for } n \geq 1. \quad (2.25)$$

In particular, one can define a *gap velocity function*  $\dot{g}$  as

$$\dot{g} = [\dot{\varphi}^1(\mathbf{X}) - \dot{\varphi}^2(\mathbf{Y}(\mathbf{X}))] \cdot \boldsymbol{\nu} + [\varphi^1(\mathbf{X}) - \varphi^2(\mathbf{Y}(\mathbf{X}))] \cdot \dot{\boldsymbol{\nu}}. \quad (2.26)$$

Since differentiating the unit normal vector  $\boldsymbol{\nu}$  in time leads to a vector which is perpendicular to it, we then have

$$\begin{aligned} \dot{g}(\mathbf{X}, t) &= [\dot{\varphi}^1(\mathbf{X}) - \dot{\varphi}^2(\mathbf{Y}(\mathbf{X}))] \cdot \boldsymbol{\nu} \\ &= [\mathbf{V}^1(\mathbf{X}) - \mathbf{V}^2(\mathbf{Y}(\mathbf{X}))] \cdot \boldsymbol{\nu}. \end{aligned} \quad (2.27)$$

During persistent contact the gap remains constant in time, i.e.  $g = 0$ ; thus the gap velocity function  $\dot{g}$  is zero during this period of time.

Drawing an analogy with the elasto-plastic problem, as seen in LAURSEN & SIMO[20], we can express the contact problem including the persistency constraint in the following way:

$$g(\mathbf{X}, t) \geq 0, \quad (2.28)$$

$$t_N(\mathbf{X}, t) \geq 0, \quad (2.29)$$

$$t_N(\mathbf{X}, t)g(\mathbf{X}, t) = 0, \quad (2.30)$$

$$t_N(\mathbf{X}, t)\dot{g}(\mathbf{X}, t) = 0. \quad (2.31)$$

Equations 2.28-2.30 are the Kuhn-Tucker complementary conditions and equation 2.31 is the persistency condition. Thus, equations 2.28-2.30 reflect the impenetrability constraint, the compressive normal traction constraint and the requirement that the pressure be nonzero only when contact occurs (i.e. when  $g = 0$ ). The persistency condition states then that the rate of separation between the two surfaces will be nonzero only when the contact pressure vanishes.

## 2.4 The frictional problem

Throughout this work, we consider the Coulomb friction model with no evolution or rate dependence of the coefficient of friction  $\mu$ , although such effects may be included in the presented methodology with slight modifications (see e.g. WRIGGERS & STEIN[37]).

### 2.4.1 Contact kinematics

A mathematical description of the relative motion between two bodies in contact is essential for the consistent statement of the law governing the friction phenomena. Below, we shall describe the necessary components of contact kinematics, also given in WRIGGERS & STEIN[37] and LAURSEN & SIMO[19]. Through the definition of the gap function  $g$  by means of the closest point projection, we define a convected basis along the master surface which enables us to express the frictional constraints. Parametrizations for  $\Gamma^2$  and  $\gamma^2$  are shown in Figure 2.3.

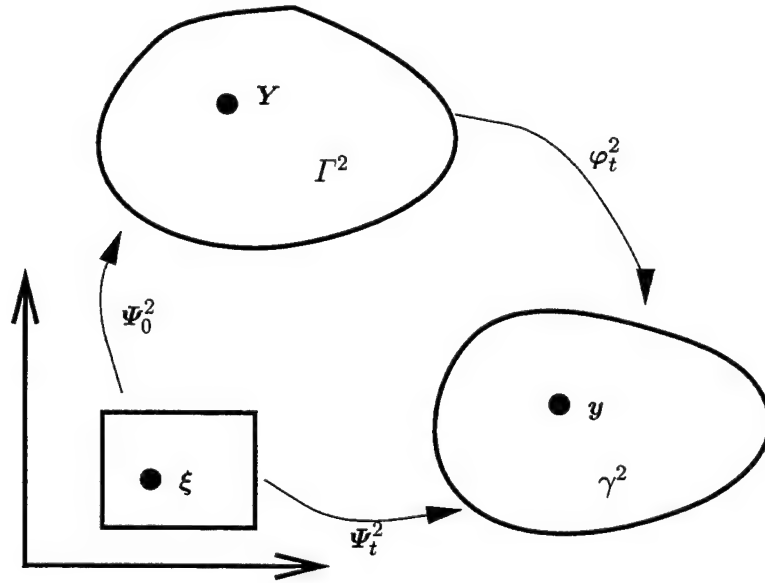


Figure 2.3: Parametrization of the contact surfaces  $\Gamma^2$  and  $\gamma^2$ .

The convected bases for  $\Gamma^2$  and  $\gamma^2$ , that is, the tangent vectors in the reference and current configuration, are defined via partial derivatives with respect to the parametrization variable

$\xi$  in the parent domain as

$$T_\alpha(\xi) := \Psi_{0,\alpha}^2(\xi) \quad (2.32)$$

and

$$\tau_\alpha(\xi) := \Psi_{t,\alpha}^2(\xi) = F_t^2(\Psi_0^2(\xi)) E_\alpha(\xi) , \quad (2.33)$$

respectively, for  $\alpha = 1, 2$ , where  $F_t^2$  is the deformation gradient corresponding to the deformation  $\varphi^2(X)$ . In equations 2.32 and 2.33 the expression  $(\cdot)_{,\alpha}$  means the partial derivative with respect to  $\xi_\alpha$ , where  $\alpha = 1, 2$ . In a two dimensional setting, the tangent plane is one dimensional, hence  $\alpha$  is omitted.

For any point  $X \in \Gamma^1$ , we obtain a corresponding point  $Y \in \Gamma^2$  through the closest point projection minimization indicated in equation 2.18, that is  $Y = \bar{Y}(X)$ . This calculated contact point is expressed in a parametrized form for the reference and current configurations as

$$\bar{Y}(X, t) = \Psi_0^2(\bar{\xi}(X, t)) \quad (2.34)$$

and

$$\bar{y}(X, t) = \Psi_t^2(\bar{\xi}(X, t)) , \quad (2.35)$$

respectively, for  $\alpha = 1, 2$ .

We denote by  $M_{\alpha\beta}$  and  $m_{\alpha\beta}$  the components of the associated positive definite metrics for the reference and current configurations, respectively, and their expressions are given by

$$M_{\alpha\beta} := T_\alpha \cdot T_\beta \quad (2.36)$$

and

$$m_{\alpha\beta} := \tau_\alpha \cdot \tau_\beta, \quad (2.37)$$

for  $\alpha, \beta = 1, 2$ . As mentioned above, for a two dimensional contact problem these metrics become positive scalars.

The friction phenomenon is ruled by the amount of relative motion between the surfaces. Thus, to measure the amount of relative slip, we need to calculate the time derivative of the material contact point  $Y \in \Gamma^2$ , the relative slip velocity, obtained through the closest point projection, i.e.,

$$\dot{Y} = \dot{\xi}_\alpha T_\alpha, \quad (2.38)$$

expressed in terms of the basis in the reference configuration. By taking the time derivative of the expression

$$\varphi^1(X) - \varphi^2(\bar{Y}(X)) = g\nu, \quad (2.39)$$

we obtain

$$[[V]] - F^2 \dot{Y} = \dot{g}\nu - g \left[ V_{,\alpha}^2 + \tau_{\alpha,\beta} \dot{\xi}^\beta \right] \cdot \tau^\alpha, \quad (2.40)$$

where the relative material velocity is expressed as

$$[[V]] := V^1(X) - V^2(\bar{Y}(X)), \quad (2.41)$$

with  $V^{(i)} := \frac{d\varphi^{(i)}}{dt}$  for  $i = 1, 2$ . In general, we use  $[[\cdot]]$  to express the relative difference of the variable  $(\cdot)$  evaluated at the two contact points  $X$  and  $Y$ . The normal and tangent components of expression 2.40 are, respectively,

$$\dot{g} = [[V]] \cdot \nu \quad (2.42)$$

and

$$A_{\alpha\beta}\dot{\xi}^\beta = \llbracket V \rrbracket \cdot \tau_\alpha + gV_{,\alpha}^2 \cdot \nu, \quad (2.43)$$

where the symmetric matrix  $A$  has the following expression:

$$A_{\alpha\beta} := m_{\alpha\beta} - g\tau_{\alpha,\beta} \cdot \nu. \quad (2.44)$$

The matrix  $A$  is assumed to be invertible at all times so that equation 2.43 defines the slip rate  $\dot{\xi}^\beta$  uniquely. Notice that, in the limit when  $g = 0$ , this property follows immediately from the positive definiteness of the metric  $m_{\alpha\beta}$ . Since it is this particular limit case which we are trying to enforce throughout the problem, the previous assumption is not too restrictive.

### 2.4.2 Coulomb friction

The friction contribution to the surface traction due to contact is expressed as follows:

$$t_T = -t_{T\beta}\tau^\beta. \quad (2.45)$$

Coulomb friction is described by means of a *slip surface* defined by the following *slip function*  $\phi$ :

$$\phi := \|t_T\| - \mu t_N. \quad (2.46)$$

The behaviour, under Coulomb friction, is characterized by a perfect stick condition (static friction) until the value of the slip function  $\phi$  becomes zero due to the increase of the tangential traction or to the decrease of the normal traction at that point. Thereafter, frictional slip (dynamic friction) occurs where the slip is related to the tangential traction (see e.g. LAURSEN & SIMO[20]), as follows:

$$v_T = \gamma \frac{t_T}{\|t_T\|}. \quad (2.47)$$

The slip can also be defined by means of kinematic considerations (see e.g. LAURSEN & SIMO[20]), as follows:

$$\begin{aligned} v_T &:= \frac{d}{dt} (\varphi^2(\bar{Y}(X))) \\ &= M_{\alpha\beta} \dot{\xi}^\beta \tau^\alpha. \end{aligned} \tag{2.48}$$

The frictional behaviour can then be described using Kuhn-Tucker complementary conditions and the persistency condition:

$$\phi = \|t_T\| - \mu t_N \leq 0, \tag{2.49}$$

$$\gamma \geq 0, \tag{2.50}$$

$$\gamma \phi = 0, \tag{2.51}$$

$$\gamma \dot{\phi} = 0, \tag{2.52}$$

where the Euclidean norm in  $\mathbb{R}^{n_{dim}}$  is given by  $\|t_T\|^2 = m^{\alpha\beta} t_{T\alpha} t_{T\beta}$  LAURSEN & SIMO[20].

**Remark 2.2** As recognized in MICHALOWSKY[22], the slip law expressed in equation 2.47 is non-associative due because the slip potential, whose gradient defines the slip direction, is not the same as the slip function  $\phi$ . If these were the same, the slip would have a dilatational component due to the dependency of  $\phi$  on the normal contact pressure  $t_N$ .

In order to approximate the perfect stick condition until the tangential traction reaches a certain value proportional to the normal pressure, we perform a penalty regularization of equation 2.47 (see e.g. WRIGGERS & STEIN [37] and LAURSEN & SIMO[20]). This regularization is achieved through the introduction of the time rate of change of the frictional traction, in a basis which must be suitably defined.

Frame indifference is achieved through the choice of the convected basis in which all expressions are written. This property will continue to hold if the rate of change is expressed using the spatial Lie derivative of the frictional traction:

$$\mathcal{L}t_T := \dot{t}_T \tau^\beta. \quad (2.53)$$

We introduce the (large) tangential penalty parameter  $\kappa_T$  for the regularization of the frictional response as follows:

$$\phi := \|t_T\| - \mu t_N \leq 0, \quad (2.54)$$

$$v_T = \gamma \frac{t_T}{\|t_T\|} + \frac{1}{\kappa_T} \mathcal{L}t_T, \quad (2.55)$$

$$\gamma \geq 0, \quad (2.56)$$

$$\gamma \phi = 0. \quad (2.57)$$

Using the Lie derivative in the regularization permits the system of equations 2.54-2.57 to be solved in closed form as in the elastoplastic problem, with the tangential penalty parameter  $\kappa_T$  playing the role of the elastic modulus. The above statement of the problem allows some slip to occur before frictional slip initiates. The amount of premature slip depends on the magnitude of the tangent penalty parameter  $\kappa_T$  and will vanish as  $\kappa_T \rightarrow \infty$ . One can make two points regarding the analogy between frictional and elastoplastic response:

1. The tangential penalty is used to preclude elastic tangential displacement, while in elastoplasticity it is used to regularize rigid-plastic response.
2. It has been suggested that the tangential stiffness  $\kappa_T$  is related somehow to the elastic stiffness of contacting asperities, so that there is a physical meaning to the analogy with an elastic modulus.



# Chapter 3

## The Weak Formulation

### 3.1 Introduction

In this chapter, the local equations presented in Chapter 2 are incorporated into the global variational formulation. As in LAURSEN & SIMO[20], the weak form of the initial boundary value problem includes the local governing equations and their associated kinematic quantities directly so that one obtains a variational principle for the two body problem that is a geometrically exact statement of the contact virtual work.

We also make use of the global statement to investigate the properties of the underlying continuum problem, by studying the behaviour of quantities such as linear momenta, angular momentum and total energy of the system. This provides us with the guidelines to later develop the algorithmic counterpart of the weak form so that it inherits its physical properties.

### 3.2 Weak form of the governing equations

The weak form of the momentum balance equations can be expressed as follows:

$$\begin{aligned}
 \sum_i \int_{\Omega^{(i)}} \rho^{(i)} \dot{V}^{(i)} \delta \varphi^{(i)} d\Omega + \sum_i \int_{\Omega^{(i)}} \mathbf{P}^{(i)} : \text{Grad}(\delta \varphi^{(i)}) d\Omega \\
 = \sum_i \int_{\Omega^{(i)}} \mathbf{b}^{(i)} \cdot \delta \varphi^{(i)} d\Omega + \sum_i \int_{\Gamma_\sigma^{(i)}} \bar{\mathbf{t}} \cdot \delta \varphi^{(i)} d\Gamma \\
 + \int_{\cup \Gamma_c} [t_N \delta g - t_{T\beta} \delta \xi^\beta] d\Gamma
 \end{aligned} \quad (3.1)$$

for all admissible variations  $\delta \varphi^{(i)}$  ( $i = 1, 2$ ) satisfying

$$\delta \varphi^{(i)} \in \left\{ \eta : \Omega^{(i)} \rightarrow \mathbb{R}^{n_{dim}} \mid \eta|_{\Gamma_\sigma^{(i)}} = \mathbf{0} \right\}. \quad (3.2)$$

The terms on the right-hand side of equation 3.1 represent the work done by the body forces  $\mathbf{b}$ , the surface tractions  $\bar{\mathbf{t}}$  and the contact tractions  $\tilde{\mathbf{t}}$ .

By manipulating the definition of the gap function  $g$  as expressed in equation 2.17, we obtain the expression for  $\delta g$ , so that by replacing the time derivative in equation 2.42 with the variation of  $\varphi$ , we obtain

$$\begin{aligned}
 \delta g &= \llbracket \varphi \rrbracket \cdot \boldsymbol{\nu} \\
 &= [\delta \varphi^1(\mathbf{X}) - \delta \varphi^2(\bar{\mathbf{Y}}(\mathbf{X}))] \cdot \boldsymbol{\nu}.
 \end{aligned} \quad (3.3)$$

The same replacement can be done in equation 2.43 to find  $\delta \xi^\alpha$  for  $\alpha = 1, 2$ , yielding the following expression:

$$\begin{aligned}
 A_{\alpha\beta} \delta \xi^\beta &= \llbracket \delta \varphi \rrbracket \cdot \boldsymbol{\tau}_\alpha + g \varphi_{,\alpha}^2 \cdot \boldsymbol{\nu} \\
 &= [\delta \varphi^1(\mathbf{X}) - \delta \varphi^2(\bar{\mathbf{Y}}(\mathbf{X}))] \cdot \boldsymbol{\tau}_\alpha + g \delta \varphi_{,\alpha}^2 \cdot \boldsymbol{\nu},
 \end{aligned} \quad (3.4)$$

where  $A_{\alpha\beta}$  has the same definition as in equation 2.44. Notice that during contact, the solution has to satisfy  $g = 0$ , hence we can rewrite equation 3.4 as

$$\delta \xi^\alpha = [\delta \varphi^1(\mathbf{X}) - \delta \varphi^2(\bar{\mathbf{Y}}(\mathbf{X}))] \cdot \tau^\alpha . \quad (3.5)$$

So far we have followed closely the geometric structure of the contact constraints and have not made any assumptions which are not valid for the continuum.

### 3.3 Properties of the weak formulation

In this section, we study the evolution of various physical quantities associated with the mechanical system. In particular, we observe that the frictionless contact problem among elastic bodies allowing finite deformations is a unilaterally constrained infinite dimensional Hamiltonian system which leads to the conservation laws described below.

Consider the total linear momentum,

$$\mathbf{L} := \sum_i \int_{\Omega^{(i)}} \rho^{(i)} \mathbf{V}^{(i)} d\Omega , \quad (3.6)$$

and the total angular momentum,

$$\mathbf{J} := \sum_i \int_{\Omega^{(i)}} \boldsymbol{\varphi}^{(i)} \times \rho^{(i)} \mathbf{V}^{(i)} d\Omega , \quad (3.7)$$

of a pair of solids ( $i = 1, 2$ ). The symbol  $\times$  refers to the cross product of two vectors in  $\mathbb{R}^3$  or the equivalent definition if we work in a two dimensional formulation. Similarly, we consider the total energy of the system,

$$\begin{aligned} \mathcal{E} &:= \sum_i \int_{\Omega^{(i)}} \frac{1}{2} \rho^{(i)} \mathbf{V}^{(i)} \cdot \mathbf{V}^{(i)} d\Omega + \sum_i \int_{\Omega^{(i)}} W^{(i)} d\Omega \\ &= \mathcal{K} + \mathcal{W} , \end{aligned} \quad (3.8)$$

where the total kinetic energy of the system is denoted by  $\mathcal{K}$  and the total elastic strain energy of the system is denoted by  $\mathcal{W}$ .

The case of interest in this analysis is the homogeneous Neumann problem, for which neither boundary displacements nor external loading is imposed; consequently, the linear and angular momenta are conserved (i.e.  $L = \text{constant}$  and  $J = \text{constant}$ ).

For the frictionless problem,  $t_T = 0$ , the linear momentum  $L$ , the angular momentum  $J$  and the total energy of the system  $\mathcal{E}$  are conserved (i.e. these quantities are constants of motion). When friction is present, energy dissipates from the system. These conservation properties are investigated in the next section.

### 3.3.1 Conservation properties of contact

#### 3.3.1.1 Conservation of linear momentum

Since no boundary displacements are imposed in this case,  $\Gamma_\varphi^{(i)} = \emptyset$  and we can use a constant as an admissible variation, as follows:

$$\delta\varphi^{(i)} := \mathbf{a}, \quad (3.9)$$

for  $i = 1, 2$ , with  $\mathbf{a} \in \mathbb{R}^{n_{dim}}$  constant. For this case, then,  $\text{Grad}(\delta\varphi^{(i)}) \equiv 0$ . Using equation 3.1 together with the definition 3.6 and taking into account that  $\bar{\mathbf{t}} = \mathbf{0}$  and  $\mathbf{b} = \mathbf{0}$ , we can state the following:

$$\begin{aligned} \mathbf{a} \cdot \frac{dL}{dt} &= \sum_i \int_{\Omega^{(i)}} \rho^{(i)} \dot{\mathbf{v}}^{(i)} \cdot \mathbf{a} \, d\Omega \\ &= \int_{\cup \Gamma_c} \bar{\mathbf{t}} \cdot [\mathbf{a} - \mathbf{a}] \, d\Gamma \\ &= 0, \forall \mathbf{a} \in \mathbb{R}^{n_{dim}}. \end{aligned} \quad (3.10)$$

Consequently,  $\frac{dL}{dt} = \mathbf{0}$ .

## 3.3.1.2 Conservation of angular momentum

Let us consider the following admissible variations:

$$\delta\varphi^{(i)}(X) := w \times \varphi^{(i)}, \quad (3.11)$$

for  $i = 1, 2$ , with  $w \in \mathbb{R}^{n_{dim}}$  constant, and  $x^{(i)} = \varphi^{(i)}(X)$ . We then obtain

$$\text{Grad}(\delta\varphi^{(i)}) = \hat{W}F^{(i)}, \quad (3.12)$$

where  $\hat{W}$  is the skew symmetric tensor cooresponding to the axial vector  $w$ , that is,

$$\hat{W}a = w \times a \quad \forall a \in \mathbb{R}^{n_{dim}}. \quad (3.13)$$

Substituting the variation defined in equation 3.11 into equation 3.1, and using the definition 3.7 and the equality 3.12, we obtain the following expression:

$$\begin{aligned} w \cdot \frac{dJ}{dt} &= \sum_i w \cdot \int_{\Omega^{(i)}} [\dot{x}^{(i)} \times \rho^{(i)} V^{(i)} + x^{(i)} \times \rho^{(i)} \dot{V}^{(i)}] d\Omega \\ &= w \cdot \sum_i \int_{\Omega^{(i)}} x^{(i)} \times \rho^{(i)} \dot{V}^{(i)} d\Omega \\ &= \sum_i \int_{\Omega^{(i)}} \rho^{(i)} \dot{V}^{(i)} \cdot (w \times x^{(i)}) d\Omega. \end{aligned} \quad (3.14)$$

Also, for this particular variation  $\delta g$  takes the following form:

$$\begin{aligned} \delta g &= \nu \cdot [w \times (\varphi^1(X) - \varphi^2(Y(X)))] \\ &= g\nu \cdot [w \times \nu] \\ &= 0 \quad \forall w \in \mathbb{R}^{n_{dim}}. \end{aligned} \quad (3.15)$$

where we have used the definition of the gap function  $g$  in 2.17. For this choice of variation, the expression  $\delta\xi^\beta$  takes the following form:

$$\begin{aligned} A_{\alpha\beta}\delta\xi^\beta &= \tau_\alpha \cdot [w \times (\varphi^1(X) - \varphi^2(Y(X)))] + g\nu \cdot [w \times \varphi_{,\alpha}^2] \\ &= g\tau_\alpha \cdot [w \times \nu] + g\nu \cdot [w \times \tau_\alpha] \\ &= 0 \quad \forall w \in \mathbb{R}^{n_{dim}}, \end{aligned} \tag{3.16}$$

where we have once more used the definition of the gap function  $g$  in equation 2.17. Using these last two results, i.e. equations 3.15 and 3.16, we can express the variation of the angular momentum as follows:

$$\begin{aligned} w \cdot \frac{dJ}{dt} &= - \int_{\cup \Omega^{(i)}} P^{(i)} : \hat{W} F^{(i)} d\Omega \\ &= 0 \quad \forall w \in \mathbb{R}^{n_{dim}}, \end{aligned} \tag{3.17}$$

where we have used the symmetry property of the Kirchhoff stress tensor, detailed in equation 2.6. Consequently,  $\frac{dJ}{dt} = 0$ .

### 3.3.1.3 Evolution of the energy

The evolution of the energy can be obtained by choosing the following variation:

$$\delta\varphi^{(i)} = V^{(i)} \quad \text{for } i = 1, 2. \tag{3.18}$$

Substituting this variation into equation 3.1 and using the definition 3.8, we obtain the following expression:

$$\begin{aligned}
 \frac{d\mathcal{E}}{dt} &= \sum_i \int_{\Omega^{(i)}} \rho^{(i)} \dot{\mathbf{v}}^{(i)} \cdot \mathbf{v}^{(i)} d\Omega + \sum_i \int_{\Omega^{(i)}} \frac{\partial W^{(i)}}{\partial \mathbf{F}^{(i)}} : \text{Grad}(\mathbf{v}^{(i)}) d\Omega \\
 &= \int_{\cup \Gamma_c} [t_N \dot{g} - t_{T\beta} \dot{\xi}^\beta] d\Gamma \\
 &= - \int_{\cup \Gamma_c} t_{T\beta} \dot{\xi}^\beta d\Gamma,
 \end{aligned} \tag{3.19}$$

where we have used the definition of the gap velocity function  $\dot{g}$  expressed in equation 2.42 and the persistency condition referred to in equation 2.31. From expression 3.19, we can observe that any change in the energy of the system comes from the frictional term. Energy is dissipated only when frictional slip occurs, because of the condition  $\gamma\phi = 0$  in 2.57. That is, if  $\phi < 0$ , then,  $\gamma = 0$ . Using this last result in equations 2.47 and 2.48, we find that  $\dot{\xi} = 0$ . When frictional slip occurs,  $\phi = 0$  due to condition 2.57; therefore, combining equations 2.47 and 2.48 yields an expression for  $\dot{\xi}$ , given by

$$\dot{\xi}^\alpha = \frac{\gamma}{\|t_T\|} M^{\alpha\beta} t_{T\beta} \tag{3.20}$$

Substituting equation 3.20 into equation 3.19, and observing that  $\|t_T\| = \mu t_N$ , we find

$$\begin{aligned}
 \frac{d\mathcal{E}}{dt} &= - \int_{\cup \Gamma_c} \mu t_N \gamma \left( \frac{\|t_T\|_{ref}}{\|t_T\|} \right)^2 d\Gamma \\
 &\leq 0,
 \end{aligned} \tag{3.21}$$

where

$$\|t_T\|_{ref}^2 := t_{T\alpha} M^{\alpha\beta} t_{T\beta}. \tag{3.22}$$

We have used equations 2.24 and 2.56 to establish the sign of the integrand.

We denote by  $\mathcal{D}_{fric}$ , the energy dissipation introduced by the friction phenomena:

$$\begin{aligned}
 \mathcal{D}_{fric} &:= \int_{\cup \Gamma_c} \mu t_N \gamma \left( \frac{\|t_T\|_{ref}}{\|t_T\|} \right)^2 d\Gamma \\
 &\geq 0.
 \end{aligned} \tag{3.23}$$

Consequently,  $\frac{d\mathcal{E}}{dt} = -\mathcal{D}_{fric}$ .

**Remark 3.1** Notice that in the absence of friction, namely  $\mu = 0$ , the energy of the system is conserved (i.e.  $\frac{d\mathcal{E}}{dt} = 0$ ).



# Chapter 4

## Finite Element Implementation

### 4.1 Introduction

The finite element method is applied to discretize the equations delineated in the previous chapters, including the contribution of the contact forces, though the actual expression of the normal contact pressure and tangential traction is postponed until Chapters 5 and 6).

The finite element implementation developed in this chapter is equivalent to those presented by WRIGGERS & SIMO [36] for the two dimensional case and by PARISCH[27] for the three dimensional case. This section is based particularly on the work done by LAURSEN AND SIMO [20], who developed a general methodology for isoparametric elements in two and three dimensions.

### 4.2 Time discretization of the weak equation

We perform a discretization of the time interval of interest  $[0, \infty]$  into subintervals  $[t_n, t_{n+1}]$ . We denote by  $\Delta t > 0$ , the corresponding *time step*  $\Delta t = t_{n+1} - t_n$ , where we assume that  $t_n = n\Delta t$ . For a variable  $g(t)$  continuous in time, we perform an algorithmic approximation:

$$g_n \approx g(n\Delta t) = g(t_n) \quad (4.1)$$

and

$$g_{n+\frac{1}{2}} = \frac{1}{2} (g_n + g_{n+1}) . \quad (4.2)$$

In the absence of external forces and imposed displacements, equation 3.1 is discretized in time as follows:

$$\begin{aligned} \sum_i \int_{\Omega^{(i)}} \rho^{(i)} \frac{1}{\Delta t} \left( \mathbf{v}_{n+1}^{(i)} - \mathbf{v}_n^{(i)} \right) \cdot \delta \boldsymbol{\varphi}^{(i)} d\Omega + \sum_i \int_{\Omega^{(i)}} \mathbf{P}^{n+\frac{1}{2},(i)} : \text{Grad}(\delta \boldsymbol{\varphi}^{(i)}) d\Omega \\ = \int_{\cup \Gamma_c} [t_N \delta g - t_{T\beta} \xi^\beta] d\Gamma , \end{aligned} \quad (4.3)$$

for all admissible variations  $\delta \boldsymbol{\varphi}^{(i)}$ ,  $i = 1, 2$ .

Following the work of SIMO & TARNOW [30], the internal elastic forces can be time discretized in such a way as to inherit all the conserving properties inherent to the unconstrained continuum problem. The algorithm, which allows finite deformations, is a second order conserving approximation of the internal force term in equation 3.1 for the elastodynamic problem, and is stated as follows:

$$\int_{\Omega^{(i)}} \mathbf{P}^{n+\frac{1}{2},(i)} : \text{Grad}(\delta \boldsymbol{\varphi}^{(i)}) d\Omega = \int_{\Omega^{(i)}} \nabla_{n+\frac{1}{2}}(\delta \boldsymbol{\varphi}) : \boldsymbol{\tau}^{(n+\frac{1}{2})} d\Omega , \quad (4.4)$$

where the Kirchhoff stress  $\boldsymbol{\tau}$  at time  $t_{n+\frac{1}{2}}$  is expressed using the Saint-Venant Kirchhoff model detailed in equation 2.9:

$$\boldsymbol{\tau}^{(n+\frac{1}{2})} := \mathbf{F}_{n+\frac{1}{2}} \left( \frac{1}{2} C(\mathbf{E}_n + \mathbf{E}_{n+1}) \right) \mathbf{F}_{n+\frac{1}{2}}^T . \quad (4.5)$$

The deformation gradient tensor  $\mathbf{F}_{n+\frac{1}{2}}$  is computed at the mid-point configuration  $\boldsymbol{\varphi}_{n+\frac{1}{2}}$ , and is given as follows:

$$\mathbf{F}_{n+\frac{1}{2}} := \text{Grad} \varphi_{n+\frac{1}{2}}. \quad (4.6)$$

The Green-Lagrange strain tensors  $\mathbf{E}_{n+1}$  and  $\mathbf{E}_n$  are computed at the configurations corresponding to  $t_{n+1}$  and  $t_n$ , respectively.

Thus, we can rewrite equation 4.3 as

$$\begin{aligned} \sum_i \int_{\Omega^{(i)}} \rho^{(i)} \frac{1}{\Delta t} \left( \mathbf{v}_{n+1}^{(i)} - \mathbf{v}_n^{(i)} \right) \cdot \delta \varphi^{(i)} d\Omega &+ \sum_i \int_{\Omega^{(i)}} \nabla_{n+\frac{1}{2}}(\delta \varphi^{(i)}) : \frac{1}{2} \mathbf{C}(\mathbf{E}_{n+1} + \mathbf{E}_n) d\Omega \\ &= \int_{\cup \Gamma_c} t_N \delta g - t_{T\beta} \xi^\beta d\Gamma, \end{aligned} \quad (4.7)$$

combined with the following standard time stepping scheme:

$$\frac{1}{\Delta t} (\varphi_{n+1} - \varphi_n) = \mathbf{V}_{n+\frac{1}{2}}. \quad (4.8)$$

### 4.3 Conserving properties of the algorithm

In SIMO & TARNOV [30], this algorithm is shown to have important conserving properties, which we detail below. In the following proofs, we assume that we have a homogeneous Neumann problem, i.e. that no displacements are imposed on the body ( $\Gamma_\varphi = \emptyset$ ) and no external body forces or surface tractions are applied to the body ( $\mathbf{b} = \mathbf{0}$  and  $\bar{\mathbf{t}} = \mathbf{0}$ ). Most importantly, we assume that the problem is unconstrained, that is that no contact forces are present as yet in the problem ( $\Gamma_c = \emptyset$ ).

#### 4.3.1 Internal linear momentum

In the time discretized setting, conservation of linear momentum holds under the above assumptions and is expressed in the following way:

$$L_{n+1} = L_n , \quad (4.9)$$

where the total linear momentum is defined by

$$L_t := \sum_i \int_{\Omega^{(i)}} \rho^{(i)} \mathbf{v}_t^{(i)} d\Omega. \quad (4.10)$$

Using the expression 4.7 and choosing  $\delta\varphi^{(i)} = \mathbf{a} \in \mathbb{R}^{n_{dim}}$ , where  $\mathbf{a}$  is any constant,

$$\begin{aligned} [L_{n+1} - L_n] \cdot \mathbf{a} &= \sum_i \int_{\Omega^{(i)}} \rho^{(i)} (\mathbf{v}_{n+1} - \mathbf{v}_n) \cdot \mathbf{a} d\Omega \\ &= -\Delta t \int_{\cup \Omega^{(i)}} \nabla_{n+\frac{1}{2}}(\mathbf{a}) : \boldsymbol{\tau}^{(n+\frac{1}{2})} d\Omega \\ &= \mathbf{0} , \end{aligned} \quad (4.11)$$

since  $\nabla_{n+\frac{1}{2}}(\mathbf{a}) = 0$ . We conclude that there is no contribution of the internal forces to the linear momentum  $L_t$ .

### 4.3.2 Internal angular momentum

In the time discretized setting, conservation of angular momentum holds under the above assumptions and is expressed in the following way:

$$\mathbf{J}_{n+1} = \mathbf{J}_n , \quad (4.12)$$

where the total angular momentum of the system is defined by

$$\mathbf{J}_t = \sum_i \int_{\Omega^{(i)}} \rho^{(i)} \boldsymbol{\varphi}_t^{(i)} \times \mathbf{v}_t^{(i)} d\Omega . \quad (4.13)$$

We can reorganize the previous expression in the following way:

$$\begin{aligned}
 J_{n+1} - J_n &= \sum_i \int_{\Omega^{(i)}} \left[ \varphi_{n+1}^{(i)} \times \rho^{(i)} \mathbf{V}_{n+1}^{(i)} - \varphi_n^{(i)} \times \rho^{(i)} \mathbf{V}_n^{(i)} \right] d\Omega \\
 &= \sum_i \int_{\Omega^{(i)}} \left[ \varphi_{n+\frac{1}{2}}^{(i)} \times \rho^{(i)} \left( \mathbf{V}_{n+1}^{(i)} - \mathbf{V}_n^{(i)} \right) \right. \\
 &\quad \left. + \left( \varphi_{n+1}^{(i)} - \varphi_n^{(i)} \right) \times \rho^{(i)} \mathbf{V}_{n+\frac{1}{2}}^{(i)} \right] d\Omega .
 \end{aligned} \tag{4.14}$$

Using equation 4.8, we see that on the last term of the integrand of equation 4.14 vanishes due to co-linearity. We also use the following admissible variation:

$$\delta \varphi^{(i)} = \varphi_{n+\frac{1}{2}}^{(i)} \times w , \tag{4.15}$$

with  $w \in \mathbb{R}^{n_{dim}}$  (constant) and substitute it into equation 4.7. Then, the first term of the integrand of equation 4.14 becomes

$$\begin{aligned}
 [J_{n+1} - J_n] \cdot w &= \sum_i w \cdot \int_{\Omega^{(i)}} \varphi_{n+\frac{1}{2}}^{(i)} \times \rho^{(i)} \left( \mathbf{V}_{n+1}^{(i)} - \mathbf{V}_n^{(i)} \right) d\Omega \\
 &= - \sum_i w \cdot \int_{\Omega^{(i)}} \text{Grad}(x_{n+\frac{1}{2}} \times w) F_{n+\frac{1}{2}}^{-1} : \tau^{(n+\frac{1}{2})} d\Omega \\
 &= - \sum_i w \cdot \int_{\Omega^{(i)}} \hat{W} : \tau^{(n+\frac{1}{2})} d\Omega \\
 &= 0 ,
 \end{aligned} \tag{4.16}$$

where we have used the result

$$\text{Grad} \left( x_{n+\frac{1}{2}}^A \times w \right) = \hat{W} F_{n+\frac{1}{2}} , \tag{4.17}$$

in which  $\hat{W}$  is the skew-symmetric tensor with axial vector  $w$ . As the contraction of a symmetric and a skew-symmetric tensor vanishes, we conclude that there are no contributions of the internal forces to the total angular momentum of the system.

### 4.3.3 Internal energy

In the time discretized setting, we prove that energy is conserved under the above assumptions. The energy of the system consists of the kinetic energy  $\mathcal{K}_t$  for  $t \in [t_n, t_{n+1}]$ , which can be written as

$$\mathcal{K}_t = \sum_i \int_{\Omega^{(i)}} \frac{1}{2} \rho^{(i)} V_t^{(i)} \cdot V_t^{(i)} d\Omega, \quad (4.18)$$

and the potential energy  $\mathcal{W}_t$  for  $t \in [t_n, t_{n+1}]$ , caused by the internal elastic forces, given by

$$\mathcal{W}_t = \sum_i \int_{\Omega^{(i)}} W d\Omega, \quad (4.19)$$

where  $W$  is the elastic potential as defined in equation 2.9. Using the admissible variation

$$\begin{aligned} \delta \varphi^{(i)} &= \mathbf{v}_{n+\frac{1}{2}}^{(i)} \\ &= \frac{1}{\Delta t} (\varphi_{n+1}^{(i)} - \varphi_n^{(i)}), \end{aligned} \quad (4.20)$$

we can rewrite the left-hand side of equation 4.7 in the following way:

$$\begin{aligned} \sum_i \int_{\Omega^{(i)}} \frac{1}{\Delta t} \rho^{(i)} (\mathbf{v}_{n+1}^{(i)} - \mathbf{v}_n^{(i)}) \cdot \mathbf{v}_{n+\frac{1}{2}}^{(i)} d\Omega &= \sum_i \int_{\Omega^{(i)}} \frac{1}{\Delta t} \rho^{(i)} (\|\mathbf{v}_{n+1}^{(i)}\|^2 - \|\mathbf{v}_n^{(i)}\|^2) d\Omega \\ &= \frac{1}{\Delta t} [\mathcal{K}_{n+1} - \mathcal{K}_n]. \end{aligned} \quad (4.21)$$

where  $\|\cdot\|$  denotes the standard Euclidean norm. The right-hand side of equation 4.7, when using the above variation, becomes

$$\begin{aligned} \mathcal{K}_{n+1} - \mathcal{K}_n &= - \sum_i \int_{\Omega^{(i)}} \nabla_{n+\frac{1}{2}} (\varphi_{n+1} - \varphi_n) : \tau^{(n+\frac{1}{2})} d\Omega \\ &= - \sum_i \int_{\Omega^{(i)}} [E_{n+1} - E_n] : \frac{1}{2} C [E_{n+1} + E_n] d\Omega \\ &= -(\mathcal{W}_{n+1} - \mathcal{W}_n), \end{aligned} \quad (4.22)$$

where we have used the Saint-Venant Kirchhoff model to describe the elastic behaviour of the bodies. We conclude then that the total energy  $\mathcal{E}_t = \mathcal{K}_t + \mathcal{W}_t$  is conserved.

## 4.4 The semi-discrete equations

The weak form of the governing equations of our mechanical system is discretized in space, resulting in a set of semi-discrete equations as explained in [14]. These equations are, essentially, a set of nonlinear ordinary differential equations.

The space is discretized using a standard isoparametric finite element formulation; thus equation 3.1 is transformed into the form

$$\dot{\mathbf{d}}(t) = \mathbf{M}^{-1} \mathbf{p}(t), \quad (4.23)$$

$$\dot{\mathbf{p}}(t) = -\mathbf{f}_{int}(\mathbf{d}(t)) + \mathbf{f}_c(\mathbf{d}(t)) + \mathbf{f}_{ext}, \quad (4.24)$$

where  $\mathbf{d}$  is the vector of nodal displacements,  $\mathbf{f}_{int}$  is the vector of internal forces due to elastic deformation,  $\mathbf{f}_{ext}$  represents the discretized vector of external forces such as body forces and  $\mathbf{f}_c$  is the vector of contact forces. We have introduced the vector of nodal linear momenta as an intermediate quantity which is defined in the following way:

$$\mathbf{p} := \mathbf{M} \mathbf{v}, \quad (4.25)$$

where  $\mathbf{v} := \dot{\mathbf{d}}(t)$ .

As for any initial value problem the equations 4.23 and 4.24 are solved with a set of initial conditions for  $\mathbf{d}$  and  $\dot{\mathbf{d}}$  (or  $\mathbf{p}$ ), given by

$$\mathbf{d}(0) = \mathbf{d}_0, \quad (4.26)$$

$$\dot{\mathbf{d}}(0) = \mathbf{v}_0. \quad (4.27)$$

## 4.5 Space discretization

To obtain the semi-discrete system of equations described in the previous section, let us consider the standard isoparametric finite element space discretization,

$$\mathbf{X} = \sum_{A=1}^{n_{node}} N^A(\boldsymbol{\xi}) \mathbf{X}_A \quad (4.28)$$

and

$$\boldsymbol{\varphi}^{(i)}(\mathbf{X}) = \mathbf{X} + \sum_{A=1}^{n_{node}} N^A(\boldsymbol{\xi}) \mathbf{d}_A^{(i)}, \quad (4.29)$$

where  $\mathbf{X}$  represents the reference coordinate field and  $\boldsymbol{\varphi}^{(i)}(\mathbf{X})$  represents the current coordinate field. The interpolation shape functions are expressed by  $N^A : \square \rightarrow \mathbb{R}$  with  $\boldsymbol{\xi} \in \square$  for  $A = 1, \dots, n_{node}$  (the total number of nodes in the isoparametric element). Finally, we have the vector of nodal displacements  $\mathbf{d}_A^{(i)} \in \mathbb{R}^{n_{dim}}$  and the vector of reference nodal coordinates  $\mathbf{X}_A \in \mathbb{R}^{n_{dim}}$ . The variables  $\mathbf{d}_A^{(i)}$  can be grouped into the vector  $\mathbf{d} \in \mathbb{R}^{n_{eq}}$  where  $n_{eq} = n_{dim} \times n_{node}$ .

The mass matrix  $\mathbf{M}$  is defined by the standard assembly procedure:

$$\mathbf{M} = \sum_{e=1}^{n_{el}} \mathbf{M}^e, \quad (4.30)$$

where  $\mathbf{M}^e$  is the elemental mass matrix and  $n_{el}$  is the total number of elements. For an element with  $n_{en}$  nodes, the elemental mass matrix  $\mathbf{M}^e$  is as follows:

$$\mathbf{M}^e = \begin{bmatrix} M_{11} \mathbf{1}_{n_{dim}} & \cdots & M_{1n_{en}} \mathbf{1}_{n_{dim}} \\ \vdots & \ddots & \vdots \\ M_{n_{en}1} \mathbf{1}_{n_{dim}} & \cdots & M_{n_{en}n_{en}} \mathbf{1}_{n_{dim}} \end{bmatrix}, \quad (4.31)$$

where  $\mathbf{1}_{n_{dim}}$  is the rank-two identity matrix in  $\mathbb{R}^{n_{dim}}$  and  $M_{AB}$  for a body  $i$  is defined as



$$M_{AB} = \int_{\Omega_e^{(i)}} \rho^{(i)} N^A N^B d\Omega \quad (4.32)$$

where  $\Omega_e^{(i)}$  is the elemental volume for body  $i$ . This definition of the mass matrix is called the *consistent mass matrix*. In subsequent chapters of this work, we may also use the *lumped mass matrix*, which is obtained by performing a standard row-sum technique [14], and is expressed as follows:

$$M_{AB} = M_A \delta_{AB} \quad (\text{no sum}) , \quad (4.33)$$

where  $M_A = \int_{\Omega_e^{(i)}} \rho^{(i)} N^A d\Omega$ .

The external force vector  $\mathbf{f}_{ext} \in \mathbb{R}^{n_{eq}}$  corresponds to the contributions of the body forces  $\mathbf{b}$  and the imposed external tractions  $\bar{\mathbf{t}}$  applied to  $\Gamma_\sigma$ . The internal force vector  $\mathbf{f}_{int} \in \mathbb{R}^{n_{eq}}$  corresponds to the stress divergence term in the continuum; its expression is as follows:

$$\mathbf{f}_{int} = \int_{\cup \Omega^{(i)}} \mathbf{B}_t^T \boldsymbol{\tau}_t d\Omega , \quad (4.34)$$

where  $\mathbf{B}_t$  is the standard strain operator,

$$\mathbf{B}_t \delta \mathbf{d} := \nabla_t^s \delta \mathbf{u} = \text{sym} [\text{Grad}[\delta \mathbf{u}] \mathbf{F}_t^{-1}] , \quad (4.35)$$

and  $\mathbf{u} := \boldsymbol{\varphi}_t(\mathbf{X}) - \mathbf{X}$  represents the displacement field at time  $t$ .

#### 4.5.1 Closest point projection in the space-discretized setting

To be able to define the contact force vector  $\mathbf{f}_c$  in equation 4.24, we first need to develop the concept of the closest point projection in a space discretized setting and expand on the notation involved.

We make use of the standard slave/master data structure (see [12]). We can identify a slave body and a master body whose surfaces are now composed of surface elements and

surface nodes. We perform a contact detection procedure for every surface node in the slave body, denoted by  $S$ , to check for possible penetration into the master body. A contact point  $\bar{Y}(\mathbf{X}_S)$  is established on the surface of the master body which can be identified by a location within a particular surface element defined by its nodes, i.e.  $M1, M2, \dots$ . Figure 4.1 shows a schematic view of a slave node  $S$  penetrating a master surface element and the definition of the contact point  $\bar{Y}(\mathbf{X}_S)$  by means of a closest point projection.

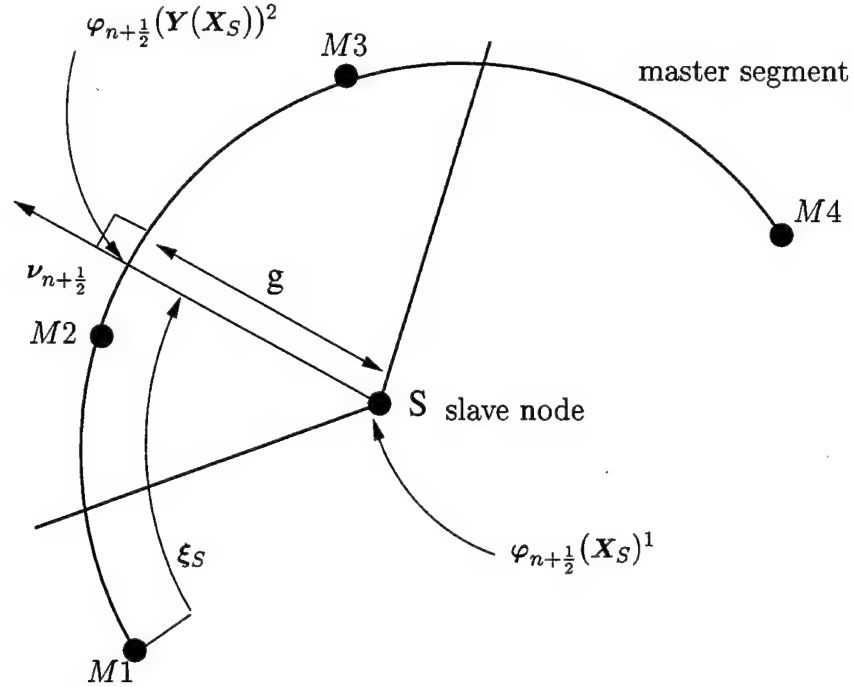


Figure 4.1: Schematic drawing of the closest point procedure on a discrete space setting in two dimensions. In the case shown, contact is detected and the master surface element is defined by 4 nodes.

The closest point projection shown in Figure 4.1 leads us to establish a *contact element* consisting of the contacting slave node  $S$  (with coordinates  $\mathbf{X}_S$ ) and the nodes that define the master segment containing the penetration point  $\bar{Y}(\mathbf{X}_S)$ . Thus, the contact element can be described by the set of nodes  $\{S, M1, M2, M3, M4, \dots\}$ .

**Remark 4.1** Throughout the evolution of the problem, the number of contact elements will vary depending on the number of slave nodes contacting the master body at any given time.

The contact force vector  $\mathbf{f}_c$  can then be expressed as the assembly of the force contributions from each contact element, as follows:

$$\mathbf{f}_c = \sum_{s=1}^{n_{slave}} \hat{\mathbf{f}}_{s,c} \quad (4.36)$$

where  $n_{slave}$  is the number of contact elements (or slave node/master segment pairs) at any given time and  $\hat{\mathbf{f}}_{s,c}$  is the contact force applied to each of the nodes belonging to the contact element. It is defined as

$$\hat{\mathbf{f}}_{s,c} = t_N \hat{\mathbf{G}}_s, \quad (4.37)$$

where

$$\hat{\mathbf{G}}_s = \begin{bmatrix} \boldsymbol{\nu}_s \\ -N^{M1}(\boldsymbol{\xi}_s)\boldsymbol{\nu}_s \\ -N^{M2}(\boldsymbol{\xi}_s)\boldsymbol{\nu}_s \\ \vdots \end{bmatrix} \in \mathbb{R}^{(1+n_{master}^s) \times n_{dim}}, \quad (4.38)$$

and  $n_{master}^s$  is the number of nodes in the master segment in contact with the slave node  $S$ . In equation 4.38,  $N^{MI}(\boldsymbol{\xi}_s)$  denotes the standard isoparametric shape function of node  $MI$  in the master segment at the point of contact  $\boldsymbol{\xi}_s$  with normal  $\boldsymbol{\nu}_s$ , obtained by the closest point projection procedure; see Figure 4.1.

The discrete counterpart of equation 2.39 is now

$$\mathbf{g}(\mathbf{X}_s)\boldsymbol{\nu} = \mathbf{x}_S - \sum_{I=1}^{n_{master}^s} N^{MI}(\boldsymbol{\xi}_s)\mathbf{x}_{MI}, \quad (4.39)$$

where  $\mathbf{x}_S = \boldsymbol{\varphi}^1(\mathbf{X}_S)$  and  $\mathbf{x}_{MI} = \boldsymbol{\varphi}^2(\mathbf{X}_S)$  are the current positions of the slave and master segment nodes, respectively. We note that the master segment shape functions satisfy the relation

$$\sum_{I=1}^{n_{master}^s} N^{MI}(\xi) = 1 \quad (4.40)$$

for any point  $\xi$  belonging to the master segment. We introduce the following simplifying notation:

$$\hat{d}_s = \begin{bmatrix} d_S \\ d_{M1} \\ d_{M2} \\ \vdots \end{bmatrix} \quad \text{and} \quad \hat{v}_s = \begin{bmatrix} v_S \\ v_{M1} \\ v_{M2} \\ \vdots \end{bmatrix}; \quad (4.41)$$

these refer to the displacement and velocity of the slave/master segment nodes, respectively. In general, we denote with  $(\cdot)_s$  the quantities that refer only to the contact element.

**Remark 4.2** As mentioned in Chapter 2, contact detection in a space discretized setting by means of a closest point projection may yield very different results HALLQUIST ET AL[12]. Figure 4.2 shows an example where choosing which surface is the slave reverses the contact state. The closest point projection will detect contact for the situation depicted on the left and will not detect contact if we reverse the choice of slave and master surface as shown in the figure on the right.

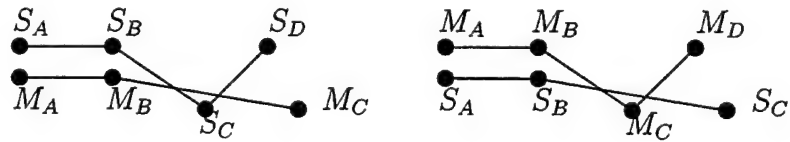


Figure 4.2: Schematic drawing of two contact situations involving with two dimensional linear surface elements. Slave nodes are denoted by  $S$  and master nodes by  $M$ .

## 4.6 Temporal and space discretization

We perform a discretization of the time interval  $\mathbb{I}$  into a series of time-steps, i.e.  $\cup_{n=0}^N [t_n, t_{n+1}]$ . The time continuity implied in expressions 4.23 and 4.24 is removed by solving the ordinary

differential equations via a time-stepping algorithm in each subinterval  $[t_n, t_{n+1}]$ . We consider a mid-point approximation to the system described by equations 4.23 and 4.24, which results in the following system:

$$\frac{1}{\Delta t}(d_{n+1} - d_n) = v_{n+\frac{1}{2}}, \quad (4.42)$$

$$\frac{1}{\Delta t}M(v_{n+1} - v_n) = -f_{int}^{(n+\frac{1}{2})} + f_c^{(n+\frac{1}{2})} + f_{ext}^{(n+\frac{1}{2})}, \quad (4.43)$$

where  $\Delta t = t_{n+1} - t_n$ , the time interval of interest. We have used the following approximations:  $d_n \approx d(t_n)$ ,  $v_n \approx v(t_n)$  and  $v_{n+\frac{1}{2}} = (v_{n+1} + v_n)/2$ . We have also eliminated the momentum from equation 4.23 by using the following approximation:

$$p(t) \approx p_t = Mv_t \quad \text{for } t \in \cup_n [t_n, t_{n+1}]. \quad (4.44)$$

The discrete form of the normal and frictional contact forces  $f_c$  are discussed and expanded upon in the next chapter. The internal force vector denoted by  $f_{int}^{(n+\frac{1}{2})}$  in equation 4.43 corresponds to the time discretization presented in SIMO & TARNOW [30]. This algorithm is a second order conserving approximation of the internal force vector  $f_{int}$  at  $t_{n+\frac{1}{2}}$  for the elastodynamic problem which allows finite deformations. The strain operator  $B_t$  defined in equation 4.35 is evaluated at the mid-point configuration  $\varphi_{n+\frac{1}{2}} := (\varphi_{n+1} + \varphi_n)/2$  and the internal force vector is then given by

$$f_{int}^{(n+\frac{1}{2})} = \sum_i \int_{\Omega^{(i)}} B_{n+\frac{1}{2}}^T \tau^{(n+\frac{1}{2})} d\Omega, \quad (4.45)$$

where the discrete Kirchhoff stress  $\tau^{(n+\frac{1}{2})}$  is calculated using the Saint-Venant Kirchhoff model defined by equation 2.9.

## Chapter 5

### Energy Restoring Momentum

### Conserving Algorithm for Frictionless Dynamic Contact

#### 5.1 Introduction

We present in this chapter the development of an *energy restoring algorithm* for frictionless dynamic contact. The objective of any algorithm is to represent the simulated physical system as closely as possible. To ensure this, the algorithmic problem should inherit the properties of the underlying continuum problem.

It is also desirable to achieve the development of a stable implicit algorithm to solve frictionless contact problems so as to obtain a robust scheme. The contact algorithm detailed here has these conserving properties, in addition to enforcing the unilateral contact constraint and the velocity constraint, as seen in ARMERO & PETŐCZ[1] and ARMERO & PETŐCZ[2].

## 5.2 Energy restoring/momentum conserving scheme

Since the goal is the conservation of energy and the momenta, let us begin by expressing the variation of the total energy in a time discretized form and analyzing the contribution of the normal contact force. The tangent component which originates from the friction phenomena is not be included in the analysis.

### 5.2.1 Time discretization of the weak form

In the absence of external forces and imposed displacements, but including the normal contribution of the contact traction, equation 4.22 takes the following form:

$$\begin{aligned} \mathcal{E}_{n+1} - \mathcal{E}_n &= (\mathcal{K}_{n+1} - \mathcal{K}_n) + (\mathcal{W}_{n+1} - \mathcal{W}_n) \\ &= \int_{\cup \Gamma_c} \tilde{\mathbf{t}}^{(n+\frac{1}{2})} \cdot [(\varphi_{n+1}^1(\mathbf{X}) - \varphi_n^1(\mathbf{X})) - \\ &\quad (\varphi_{n+1}^2(\bar{\mathbf{Y}}_{n+\theta_1}(\mathbf{X})) - \varphi_n^2(\bar{\mathbf{Y}}_{n+\theta_1}(\mathbf{X})))] d\Gamma, \end{aligned} \quad (5.1)$$

where we have left as an unknown the instant  $t_{n+\theta_1}$  at which we perform the closest point projection within the time step  $[t_n, t_{n+1}]$ . Also, in the absence of friction, the contact traction  $\tilde{\mathbf{t}}^{(n+\frac{1}{2})}$  at  $t_{n+\frac{1}{2}}$  is

$$\tilde{\mathbf{t}}^{(n+\frac{1}{2})} = t_N \boldsymbol{\nu}_{n+\theta_2}, \quad (5.2)$$

where  $\boldsymbol{\nu}_{n+\theta_2}$  is the unit normal at the contact point at the configuration at  $t_{n+\theta_2}$ . Substituting the expression of the contact force in expression 5.2 into equation 5.2, we obtain

$$\begin{aligned} \mathcal{E}_{n+1} - \mathcal{E}_n &= (\mathcal{K}_{n+1} - \mathcal{K}_n) + (\mathcal{W}_{n+1} - \mathcal{W}_n) \\ &= \int_{\cup \Gamma_c} t_N [(\varphi_{n+1}^1(\mathbf{X}) - \varphi_n^1(\mathbf{X})) - \\ &\quad (\varphi_{n+1}^2(\bar{\mathbf{Y}}_{n+\theta_1}(\mathbf{X})) - \varphi_n^2(\bar{\mathbf{Y}}_{n+\theta_1}(\mathbf{X})))] \cdot \boldsymbol{\nu}_{n+\theta_2} d\Gamma. \end{aligned} \quad (5.3)$$

We have established the following three unknowns:  $\theta_1$ ,  $\theta_2$  and  $t_N$ . We need to find expressions for these unknowns so as to satisfy the restrictions imposed by the conservation properties we want the algorithm to have, namely energy restoration and linear and angular momentum conservation.

We define the following evolution equation:

$$g_{n+1}^d := g_n^d + \nu_{n+\theta_2} \cdot \left[ (\varphi_{n+1}^1(\mathbf{X}) - \varphi_{n+1}^2(\bar{\mathbf{Y}}_{n+\theta_1}(\mathbf{X}))) - (\varphi_n^1(\mathbf{X}) - \varphi_n^2(\bar{\mathbf{Y}}_{n+\theta_1}(\mathbf{X}))) \right] . \quad (5.4)$$

In view of the above definition, we define a new gap function which we call the *dynamic gap function* and denote by  $g^d$  ARMERO & PETŐCZ [1] [2].

**Remark 5.1** Equation 5.4 is, in fact, a time discretization of the gap velocity expression, given by

$$\dot{g} = \nu \cdot [V^1(\mathbf{X}) - V^2(\bar{\mathbf{Y}}(\mathbf{X}))] . \quad (5.5)$$

Hence, the contribution of the normal contact pressure to the energy variation expressed in equation 5.1 can be rewritten as follows:

$$\mathcal{E}_{n+1} - \mathcal{E}_n = \int_{\cup \Gamma_c} t_N (g_{n+1}^d - g_n^d) d\Gamma . \quad (5.6)$$

Keeping in mind that the goal is the development of a stable algorithm in the form of an energy restoring scheme, i.e. one for which the energy is restored to the system of bodies when contact is no longer detected, we have to satisfy

$$\mathcal{E}_{n+1} - \mathcal{E}_n = -(\mathcal{P}_{n+1} - \mathcal{P}_n) , \quad (5.7)$$

where  $\mathcal{P}_t$  is, intuitively, the total regularization potential of a conservative force which will enforce the unilateral contact constraint. By construction, we then have



$$\mathcal{P}_{n+1} - \mathcal{P}_n = - \int_{\cup \Gamma_c} (U(g_{n+1}^d) - U(g_n^d)) d\Gamma_c \quad (5.8)$$

where the potential  $U$  is the individual contribution of each point in contact. Using equation 5.6 together with equation 5.8, we have the following expression for the normal contact pressure  $t_N$  ARMERO & PETŐCZ [1] [2]:

$$t_N = \begin{cases} -\frac{U(g_{n+1}^d) - U(g_n^d)}{g_{n+1}^d - g_n^d} & \text{if } g_{n+1}^d \neq g_n^d, \\ -U'(\frac{1}{2}(g_{n+1}^d + g_n^d)) & \text{if } g_{n+1}^d = g_n^d, \end{cases} \quad (5.9)$$

where  $U(g)$  is the penalty regularization potential that is used to satisfy the constraint detailed in equation 2.19.

**Remark 5.2** The regularization potential  $U(g)$  used in 5.9 can be any positive definite function which is nonzero if and only if  $g \leq 0$ . However, the function has to be such that when  $g_{n+1}^d \leq 0$  and  $g_n^d > 0$  it enforces the impenetrability constraint at  $t_{n+1}$

Intuitively, we may imagine the contact force as the sum of the action of a series of “stiff” springs located along the common surface of contact  $\cup \Gamma_c$ , where this force is a function of the existing gap between the two contacting surfaces. Armed with this physical insight, we define  $U(g)$  as follows:

$$U(g) := \begin{cases} \frac{1}{2} \kappa_N g^2 & \text{if } g \leq 0, \\ 0 & \text{if } g > 0, \end{cases} \quad (5.10)$$

where  $\kappa_N$  is a (large) penalty parameter.

The rest of the unknowns, namely  $\theta_1$  and  $\theta_2$ , are set so as to enforce the conservation of angular momentum. We add to the calculations of Section 3.3.1.2, the contribution of the normal contact force to the variation of the angular momentum. If one performs the standard substitution in the variations  $\delta\varphi$ , one obtains

$$\begin{aligned}
 [J_{n+1} - J_n] \cdot w &= \int_{\cup \Gamma_c} t_N \nu_{n+\theta_2} \cdot \left[ \varphi_{n+\frac{1}{2}}^1(X) \times w - \varphi_{n+\frac{1}{2}}^2(\bar{Y}_{n+\theta_1}(X)) \times w \right] d\Gamma \\
 &= \int_{\cup \Gamma_c} t_N \nu_{n+\theta_2} \cdot \left[ \varphi_{n+\frac{1}{2}}^1(X) - \varphi_{n+\frac{1}{2}}^2(\bar{Y}_{n+\theta_1}(X)) \right] \times w d\Gamma \\
 &= \int_{\cup \Gamma_c} t_N \left[ \nu_{n+\theta_2} \times \left( \varphi_{n+\frac{1}{2}}^1(X) - \varphi_{n+\frac{1}{2}}^2(\bar{Y}_{n+\theta_1}(X)) \right) \right] \cdot w d\Gamma. \quad (5.11)
 \end{aligned}$$

Notice that if  $\theta_1 = \theta_2 = \frac{1}{2}$ ,

$$\begin{aligned}
 [J_{n+1} - J_n] \cdot w &= \int_{\cup \Gamma_c} t_N \left[ \nu_{n+\frac{1}{2}} \times g_{n+\frac{1}{2}} \nu_{n+\frac{1}{2}} \right] \cdot w d\Gamma \\
 &= 0 \quad \forall w \in \mathbb{R}^{n_{dim}}, \quad (5.12)
 \end{aligned}$$

where we have used the gap definition in equation 2.17 at  $t_{n+\frac{1}{2}}$  and the unit normal  $\nu_{n+\frac{1}{2}}$  as defined by the closest point projection.

Conservation of linear momentum is guaranteed by construction. For completeness, we include the proof. Using the variation and the results from Section 3.3.1.1., we obtain:

$$\begin{aligned}
 [L_{n+1} - L_n] \cdot a &= \int_{\cup \Gamma_c} t_N \nu_{n+\frac{1}{2}} \cdot [a - a] d\Gamma \\
 &= 0 \quad \forall a \in \mathbb{R}^{n_{dim}}. \quad (5.13)
 \end{aligned}$$

Using the previous results, we can rewrite 5.4 as follows ARMERO & PETŐCZ[1] [2]:

$$g_{n+1}^d = g_n^d + \nu_{n+\frac{1}{2}} \cdot \left[ \left( \varphi_{n+1}^1(X) - \varphi_{n+1}^2(\bar{Y}_{n+\frac{1}{2}}(X)) \right) - \left( \varphi_n^1(X) - \varphi_n^2(\bar{Y}_{n+\frac{1}{2}}(X)) \right) \right] \quad (5.14)$$

**Remark 5.3** Equation 5.14 is a second order approximation for the evolution equation of the gap function in the continuum setting, expressed in equation 5.5.

**Remark 5.4** Notice that the definition of the dynamic gap function  $g_t^d$  is defined by the deformation  $\varphi$ , and not by the material velocities  $V$ , so that the restoring scheme can be used with only slight modifications in a quasi-static problem.

### 5.2.2 Energy restoration

The energy restoration property of our scheme for the frictionless case is stated in equation 5.7, where the total contribution of the penalty potential is a positive quantity, that is,

$$\begin{aligned}\mathcal{P}_t &:= \int_{\cup \Gamma_c} U(g_t^d) d\Gamma \\ &\geq 0.\end{aligned}\tag{5.15}$$

The combination of the energy balance expressed in equation 5.7 and the positive definiteness of the potential  $\mathcal{P}_t$  stated in equation 5.15 has two important consequences:

1. For any time step  $t_n$  in which contact is detected, i.e.  $U(g_n^d) > 0$ ,

$$\mathcal{E}_n < \mathcal{E}_0 ,\tag{5.16}$$

where  $\mathcal{E}_0$  is the initial total energy. Hence, the energy of the system of solids will never increase during the time of contact regardless of the size of the time step  $\Delta t$  used in the time marching scheme.

2. For any time step  $t_n$  in which contact is no longer detected, i.e.  $U(g_n^d) = 0$ ,

$$\mathcal{E}_n = \mathcal{E}_0 .\tag{5.17}$$

Hence, the energy of the system of solids is restored after release occurs.

We can conclude, then, that the proposed scheme is unconditionally (energy) stable.

### 5.2.3 Space discretization

Now we need to formulate equation 5.14 in a space discretized setting. The distinctive aspect is that, when in contact, we do not identify a common contact surface  $\Gamma_c$  but only a set of slave nodes  $S$  which come into contact with their master segment. As explained in Chapter 4 we have  $n_{slave}$  slave node/master segment pairs. Then, for each slave node material point  $\mathbf{X}_S$ , we can define a dynamic gap at time  $t_n$  denoted by  $g_{s,n}^d$  as follows:

$$g_{s,n+1}^d = g_{s,n}^d + \nu_{s,n+\frac{1}{2}} \cdot \left[ \left( \varphi_{n+1}^1(\mathbf{X}_S) - \varphi_{n+1}^2(\bar{\mathbf{Y}}_{n+\frac{1}{2}}(\mathbf{X}_S)) \right) - \left( \varphi_n^1(\mathbf{X}_S) - \varphi_n^2(\bar{\mathbf{Y}}_{n+\frac{1}{2}}(\mathbf{X}_S)) \right) \right] \quad (5.18)$$

With the notation introduced in the previous chapter, we can rewrite equation 5.18 in the following way:

$$g_{s,n+1}^d = g_{s,n}^d + \hat{\mathbf{G}}_{s,n+\frac{1}{2}}^T [\hat{\mathbf{d}}_{s,n+1} - \hat{\mathbf{d}}_{s,n}] , \quad (5.19)$$

for the corresponding displacements of slave and master nodes at  $t_{n+1}$  and  $t_n$ . The initialization of the dynamic gap evolution equation is achieved by using the (real) gap function  $g_{s,n}$  (i.e. the one obtained by performing a closest point projection procedure at  $t_n$ , the last time-step before contact is detected). Equation 5.18 is a second order approximation of equation 2.42 for the evolution of the real gap  $g_s$  in a space discretized formulation.

**Remark 5.5** Note that, in the one dimensional case, the dynamic gap  $g_s^d$  and the (real) gap  $g_s$  coincide, as the approximation is given by the change in the normal  $\nu_s$ .

The contribution of the normal contact force to the nodal contact vector is then given by

$$\hat{\mathbf{f}}_{s,c}^{(n+\frac{1}{2})} = t_N \hat{\mathbf{G}}_{s,n+\frac{1}{2}} , \quad (5.20)$$

where  $t_N$  is defined in equation 5.9.

### 5.2.4 Contact/release conditions

An independent part of a contact algorithm, in addition to a contact detection algorithm and the actual contact force, is the appropriate choice of a contact/release criterion. The contact/release logic developed for this scheme is presented in Table 5.1, below.

As mentioned previously, the dynamic gap  $g_{s,n}^d$  is initialized with the value of the “real” gap from the converged values of the previous time step before contact is detected at  $t_{n+1}$ . Contact is detected at  $t_{n+1}$  if  $g_{s,n+1}$  is negative, which is our indication that penetration has occurred. Subsequently, contact is detected if the dynamic gap  $g_{s,n+1}^d$  is negative. As the contact pressure  $t_N$  depends on the contact states at  $t_{n+1}$  and  $t_n$ , it only vanishes when both of these states are released states. When contact is no longer detected at  $t_{n+1}$ , we observe that  $t_N$  becomes negative; this represents the final “kick” that restores the energy back to the system of bodies.

An important aspect of our algorithm is that we enforce the unilateral constraint at  $t_{n+1}$ . At the first time step when contact is detected,  $U(g_{s,n}^d) = 0$ , so the penalty regularization is applied to enforce  $g_{n+1}^d = 0$ . Many standard contact algorithms impose the gap constraint at  $t_{n+\frac{1}{2}}$ , which leads to an oscillatory behaviour of the gap function.

## 5.3 Enforcement of the gap velocity constraint

Different mechanical contact problems involve different contact times. By contact times we mean to the period of time two bodies may remain in contact. This contact time depends on the mechanical properties of the material and the relative velocity between the bodies before collision.

In some cases, we may want to investigate the contact problem with very small time steps as we might be interested in what happens during the contact time. In other cases, we may be more interested in the collision paths and rebound velocities so we may perform the simulation with bigger time steps.

Table 5.1: Contact/release logic.

Let  $\text{cont}_{s,n}$  = contact flag at  $t_n$  (.true. or .false.), and  $g_{s,n+1}$  the (real) gap at  $t_{n+1}$  for slave node  $S$ . Then,  $\text{cont}_{s,n+1}$  is defined as follows:

```

IF (conts,n .or. (gs,n+1.le.0)) THEN
  Compute gs,n+1d using 5.18
  IF (gs,n+1d.le.0) THEN
    conts,n+1 = .true.
  ELSE
    conts,n+1 = .false.
    IF (gs,n+1.ge.0) THEN
      The dynamic gap will be initialized to the current
      gs,n+1 when evaluating 5.18 in the next time step
    ELSE
      The dynamic gap will be initialized to the current
      gs,n+1d when evaluating 5.18 in the next time step.
    ENDIF
  ENDIF
ENDIF
ELSE
  conts,n+1 = .false.
ENDIF

```

The algorithm presented works for any case including the two extreme cases detailed above. If the contact time is “big” compared to the time step chosen to perform the simulation, then the simulation falls into the category of *persistent contact*. As discussed above, during persistent contact we encounter the gap velocity constraint as stated in equation 2.31. Satisfaction of this constraint has been researched by LEE [21] and by TAYLOR & PAPADOPOULOS [33].

The goal in this section is to modify the previously presented algorithm to satisfy this additional constraint, while maintaining the conservation properties achieved so far. To this end, we restate the weak form of the balance of momentum equation presented in equation 4.3, in the following form:

$$\begin{aligned} \sum_i \int_{\Omega^{(i)}} \frac{1}{\Delta t} (\mathbf{p}_{n+1}^{(i)} - \mathbf{p}_n^{(i)}) \cdot \delta \boldsymbol{\varphi}^{(i)} d\Omega + \sum_i \int_{\Omega^{(i)}} \mathbf{P}^{n+\frac{1}{2},(i)} : \text{Grad}(\delta \boldsymbol{\varphi}^{(i)}) d\Omega \\ = \int_{\cup \Gamma_c} [t_N \delta g - t_{T\beta} \delta \xi^\beta] d\Gamma. \end{aligned} \quad (5.21)$$

Also, equation 4.8 can be rewritten as follows:

$$\frac{1}{\Delta t} (\boldsymbol{\varphi}_{n+1}^{(i)} - \boldsymbol{\varphi}_n^{(i)}) = \frac{\mathbf{p}^{(i)}}{\rho^{(i)}}, \quad (5.22)$$

for bodies  $i = 1, 2$ .

We use the following approximation to the gap velocity expression given in equation 2.27:

$$h_t = [\mathbf{V}_t^1(\mathbf{X}) - \mathbf{V}_t^2(\bar{\mathbf{Y}}_t(\mathbf{X}))] \cdot \boldsymbol{\nu}_t, \quad (5.23)$$

for any  $t \in [t_n, t_{n+1}]$ , where we have used the notation

$$h_n \approx \dot{g}(t_n) \quad (5.24)$$

We perform a penalty regularization on the linear momentum  $\mathbf{p}$  for the points belonging to the neighbourhood of the common boundary  $\mathcal{O}(\Gamma_c)$  to enforce the constraint on the gap velocity  $\dot{g}$ . Hence, we present the definition of the regularized linear momenta as follows:

$$\tilde{\mathbf{p}}_t^1 = \rho^1 \mathbf{V}_t^1 + \rho_{p,t} h_t \boldsymbol{\nu}_t \quad \forall \mathbf{X} \in \mathcal{O}(\Gamma_c), \quad (5.25)$$

$$\tilde{\mathbf{p}}_t^2 = \rho^2 \mathbf{V}_t^2 - \rho_{p,t} h_t \boldsymbol{\nu}_t \quad \forall \mathbf{Y}(\mathbf{X}) \in \mathcal{O}(\Gamma_c), \quad (5.26)$$

where  $\tilde{\mathbf{p}}_t^1$  and  $\tilde{\mathbf{p}}_t^2$  denote the modified linear momenta for each material point  $\mathbf{X}$  in the neighbourhood of the surface of body 1 and its corresponding projection  $\mathbf{Y}(\mathbf{X})$  onto body 2, respectively.

**Remark 5.6** This modification of the definition of the linear momenta in the neighbourhood of  $\Gamma_c$  leads, in the finite element implementation, to the use of a lumped mass matrix instead of a consistent mass matrix.

The penalty  $\rho_{p,t}$  is defined in the following way:

$$\rho_{p,t} := \begin{cases} m_p & \text{if } U(g_t^d) \neq 0, \\ 0 & \text{otherwise,} \end{cases} \quad (5.27)$$

where  $m_p$  is a (large) penalty parameter.

The modification to the definition of the linear momenta detailed in equations 5.25 and 5.26 and the subsequent change in the dynamic formula 5.22 remains active only while the impenetrability constraint ( $g_t^d = 0$ ) is active, as expressed in equation 5.27. Notice that for interior points, the definition of the linear momenta remains unchanged and the time updating formula 4.8 remains valid. This modification enables us to introduce a penalization on the gap velocity  $\dot{g}$  in an algorithmic way.

This enhancement to the scheme to enforce the gap velocity constraint preserves both its conservation and restoring properties. We prove this statement in the following section.

### 5.3.1 Properties of the proposed scheme

Our goal has been to enforce the constraint on the gap velocity, as expressed in equation 2.31, while maintaining the validity of the conserving properties of our scheme. Next, we inves-



tigate these properties and prove that we have created an energy restoring and momentum conserving scheme with the added characteristic of satisfying the gap velocity constraint.

**Proposition 5.1**

In the absence of external forces and imposed displacements, we have for a typical time interval  $[t_n, t_{n+1}]$ ,

1. The linear momentum is conserved, that is:

$$L_{n+1} = L_n ; \quad (5.28)$$

2. The angular momentum is conserved, that is:

$$J_{n+1} = J_n ; \quad (5.29)$$

3. The energy evolves as

$$\mathcal{E}_{n+1} + \mathcal{P}_{n+1} + \mathcal{M}_{n+1} = \mathcal{E}_n + \mathcal{P}_n + \mathcal{M}_n , \quad (5.30)$$

where

$$\begin{aligned} \mathcal{M}_t &:= \int_{\mathcal{O}(\Gamma_c)} \rho_{p,t} h_t^2 \left[ 1 + \frac{\rho_{p,t}}{2} \left( \frac{1}{\rho^1} + \frac{1}{\rho^2} \right) \right] d\Omega \\ &\geq 0 . \end{aligned} \quad (5.31)$$

**Proof:**

1. We prove that the modification to the linear momenta does not contribute to the total linear momentum of the system. Therefore, the conservation of linear momentum proved in the previous section still holds.

Consider the linear momentum of the system at time  $t$ . It is given by the sum of the linear momentum of the system at time  $t$  and the linear momentum of the system at time  $t$ .

$$\begin{aligned} \tilde{L}_t &= \sum_i \int_{\Omega^{(i)}} \rho^{(i)} \mathbf{v}_t^{(i)} d\Omega + \int_{\mathcal{O}(\Gamma_c)} \rho_{p,t} h_t (\nu - \nu) d\Omega \\ &= 0 . \end{aligned} \quad (5.32)$$

2. The conservation of the angular momentum is proved in a similar way. We first prove that the contribution to the total angular momentum of the system, by the modification to the linear momenta, vanishes. That is,

$$\begin{aligned}
 \tilde{\mathbf{J}}_t &= \sum_i \int_{\Omega^{(i)}} \varphi_t \times \rho^\alpha \mathbf{V}_t \, d\Omega + \int_{\mathcal{O}(r_c)} [\varphi_t^1(\mathbf{X}) - \varphi_t^2(\bar{\mathbf{Y}}_t(\mathbf{X}))] \times \rho_{p,t} h_t \boldsymbol{\nu}_t \, d\Omega \\
 &= \sum_i \int_{\Omega^{(i)}} \varphi_t \times \rho^\alpha \mathbf{V}_t \, d\Omega + \int_{\mathcal{O}(r_c)} g_t \boldsymbol{\nu}_t \times \rho_{p,t} h_t \boldsymbol{\nu}_t \, d\Omega \\
 &= \mathbf{J}_t,
 \end{aligned} \tag{5.33}$$

where the integrand of the surface integral vanishes due to co-linearity.

As before, the conservation of this new quantity still holds, so the conservation of the total angular momentum is confirmed.

3. In terms of the energy, we know that the following property is preserved:

$$\tilde{\mathcal{K}}_{n+1} + \mathcal{W}_{n+1} + \mathcal{P}_{n+1} = \tilde{\mathcal{K}}_n + \mathcal{W}_n + \mathcal{P}_n, \tag{5.34}$$

where we denote by  $\tilde{\mathcal{K}}_t$  the modified total kinetic energy, stated as

$$\tilde{\mathcal{K}}_t := \sum_i \int_{\Omega^{(i)}} \frac{1}{2\rho^{(i)}} \tilde{\mathbf{p}}_t^{(i)T} \tilde{\mathbf{p}}_t^{(i)} \, d\Omega, \tag{5.35}$$

for  $t \in [t_n, t_{n+1}]$ . After some algebraic manipulation, we arrive at the following expression:

$$\tilde{\mathcal{K}}_t = \mathcal{K}_t + \int_{\mathcal{O}(r_c)} \rho_{p,t} h_t^2 \left[ 1 + \frac{\rho_{p,t}}{2} \left( \frac{1}{\rho^1} + \frac{1}{\rho^2} \right) \right] \, d\Omega, \tag{5.36}$$

where

$$\begin{aligned}
 \mathcal{M}_t^h &:= \int_{\mathcal{O}(r_c)} \rho_{p,t} h_t^2 \left[ 1 + \frac{\rho_{p,t}}{2} \left( \frac{1}{\rho^1} + \frac{1}{\rho^2} \right) \right] \, d\Omega \\
 &\geq 0
 \end{aligned} \tag{5.37}$$

### 5.3.1.1 Energy restoration

The energy restoring property still holds due to equation 5.36 and the fact that the added kinetic energy  $\mathcal{M}_t$  introduced by the penalty regularization on the linear momentum is always positive and vanishes at a released state.

### 5.3.2 Finite element implementation

The previously stated modification to the definition of the linear momentum of the problem is easily implemented within a finite element formulation. To this end, we rewrite the linear momentum for a typical slave node/master segment as follows:

$$\begin{aligned}\hat{p}_{s,t} &= [\hat{M}_{s,L} + m_{s,t}\hat{G}_{s,t}\hat{G}_{s,t}^T] \hat{v}_{s,t} \\ &= \hat{M}_{s,L}\hat{v}_{s,t} + m_{s,t}h_{s,t}\hat{G}_{s,t},\end{aligned}\tag{5.38}$$

where  $t \in [t_n, t_{n+1}]$ , a typical time increment, and we denote by  $h_{s,t}$  the discrete counterpart of the gap velocity definition 2.42. It is defined as

$$\begin{aligned}h_{s,t} &:= \hat{G}_{s,t}^T \hat{v}_{s,t} \\ &= \nu_t \cdot \left[ v_t^S - \sum_{I=1}^{n_{master}^s} N^{MI}(\xi_{s,t}) v_t^{MI} \right].\end{aligned}\tag{5.39}$$

We have used in equation 5.38 the lumped mass matrix  $M_{s,L}$  for the contact element, i.e. the slave node and the master nodes that participate in the master segment. It is given by

$$\hat{M}_{s,L} := \begin{bmatrix} M_S \mathbf{1}_{n_{dim}} & & & \\ & M_{M1} \mathbf{1}_{n_{dim}} & & \\ & & M_{M2} \mathbf{1}_{n_{dim}} & \\ & & & \ddots \end{bmatrix} \in \mathbb{R}^{(1+n_{master}^s) \times (1+n_{master}^s)}, \tag{5.40}$$

to simplify the notation. In equation 5.38,  $m_{s,t}$  denotes a penalty mass added to the contacting slave and master nodes, which depends on the contact states, as follows:

$$m_{s,t} := \begin{cases} m_p & \text{if } t_{N_{s,t}} \neq 0, \\ 0 & \text{otherwise,} \end{cases} \quad (5.41)$$

where  $m_p$  is a large penalty parameter. As  $m_p \rightarrow \infty$ , the constraint  $h_{s,n+1} = 0$  for a typical interval  $[t_n, t_{n+1}]$  of contact is enforced.

Rewriting equations 4.42 and 4.43, and substituting the linear momentum  $\mathbf{p}$  for our new intermediate variable  $\hat{\mathbf{p}}$  defined in equation 5.38, we have the following:

$$\frac{1}{\Delta t} (\hat{\mathbf{d}}_{s,n+1} - \hat{\mathbf{d}}_{s,n}) = \hat{\mathbf{v}}_{s,n+\frac{1}{2}} + \frac{1}{2} \hat{\mathbf{M}}_{s,L}^{-1} (m_{s,n+1} h_{s,n+1} \hat{\mathbf{G}}_{s,n+1} + m_{s,n} h_{s,n} \hat{\mathbf{G}}_{s,n}), \quad (5.42)$$

$$\frac{1}{\Delta t} \hat{\mathbf{M}}_{s,L} (\hat{\mathbf{v}}_{s,n+1} - \hat{\mathbf{v}}_{s,n}) = -\hat{\mathbf{f}}_{s,int}^{(n+\frac{1}{2})} + \hat{\mathbf{f}}_{s,(c,mass)}^{(n+\frac{1}{2})} + \hat{\mathbf{f}}_{s,ext}^{(n+\frac{1}{2})}, \quad (5.43)$$

where we denote by  $\hat{\mathbf{f}}_{s,(c,mass)}^{(n+\frac{1}{2})}$  the contact force given by

$$\hat{\mathbf{f}}_{s,(c,mass)}^{(n+\frac{1}{2})} = \hat{\mathbf{f}}_{s,c}^{(n+\frac{1}{2})} - \hat{\mathbf{z}}_s^{(n+\frac{1}{2})}, \quad (5.44)$$

and

$$\hat{\mathbf{z}}_s^{(n+\frac{1}{2})} := \frac{1}{\Delta t} (m_{s,n+1} h_{s,n+1} \hat{\mathbf{G}}_{s,n+1} - m_{s,n} h_{s,n} \hat{\mathbf{G}}_{s,n}). \quad (5.45)$$

This added force enforces the constraint on the gap velocity, and vanishes for the nodes that are not in contact.

## 5.4 Addition of energy dissipation

Impact phenomena are difficult to simulate due to the high frequency modes present in the motion of the body during and after the impact. The need to introduce numerical

dissipation into the algorithmic scheme becomes apparent during short-term simulations where the interval in which the bodies are in contact is long compared to the time step  $\Delta t$ . To this end, we introduce a modification to the previous energy restoring algorithm. During persistent contact for the time step  $[t_n, t_{n+1}]$ ,  $U(g_{s,n+1}^d) \neq 0$  and  $U(g_{s,n}^d) \neq 0$ . Thus, the normal contact pressure  $t_N$  takes the following form for the potential function described in equation 5.10:

$$\begin{aligned} t_N &= -\kappa_N \frac{(g_{s,n+1}^d)^2 - (g_{s,n}^d)^2}{g_{s,n+1}^d - g_{s,n}^d}, \\ &= -\frac{1}{2}\kappa_N (g_{s,n+1}^d + g_{s,n}^d). \end{aligned} \quad (5.46)$$

We perform a modification of the previous expression, as follows:

$$t_N = -\kappa_N (\vartheta g_{s,n+1}^d + (1 - \vartheta)g_{s,n}^d), \quad (5.47)$$

for  $\vartheta > \frac{1}{2}$ .

**Remark 5.7** At the first and last time steps, the expression for the contact pressure remains unchanged with respect to the energy restoring one, so that we satisfy the impenetrability constraint at  $t_{n+1}$ .

We now prove that we have energy dissipation for any  $\vartheta > \frac{1}{2}$ . Let us consider the time interval  $[t_n, t_{n+1}]$  where we have persistent contact (i.e. contact is detected both at  $t_n$  and at  $t_{n+1}$ ). We have the following expression for the variation of the total energy  $\mathcal{E}_t$ :

$$\begin{aligned}
 \mathcal{E}_{n+1} - \mathcal{E}_n &= \int_{\Gamma_c} t_N (g_{n+1}^d - g_n^d) \, d\Gamma \\
 &= - \int_{\Gamma_c} \kappa_N (\vartheta g_{n+1}^d + (1 - \vartheta) g_n^d) (g_{n+1}^d - g_n^d) \, d\Gamma \\
 &= - \int_{\Gamma_c} \kappa_N \left[ \frac{1}{2} (g_{n+1}^d + g_n^d) + \left( \vartheta - \frac{1}{2} \right) (g_{n+1}^d - g_n^d) \right] (g_{n+1}^d - g_n^d) \, d\Gamma \\
 &= - \int_{\Gamma_c} (U(g_{n+1}^d) - U(g_n^d)) \, d\Gamma \\
 &\quad - \left( \vartheta - \frac{1}{2} \right) \int_{\Gamma_c} \kappa_N (g_{n+1}^d - g_n^d)^2 \, d\Gamma \\
 &= -(\mathcal{P}_{n+1} - \mathcal{P}_n) - \left( \vartheta - \frac{1}{2} \right) \int_{\Gamma_c} \kappa_N (g_{n+1}^d - g_n^d)^2 \, d\Gamma. \tag{5.48}
 \end{aligned}$$

**Remark 5.8** Notice that we revert to the energy restoring scheme when  $\vartheta = \frac{1}{2}$ .

# Chapter 6

## Frictional Dynamic Contact

### 6.1 Introduction

A relevant feature of the friction phenomenon is the presence of energy dissipation, in the form of heat. Though from the algorithmic point of view, this dissipation is advantageous (provided the algorithm inherits this dissipative property), the treatment of friction becomes complicated due to the discontinuous slip/stick behaviour of Coulomb friction.

Following the characteristic line of our research, our aim in this part of the work is to design a friction algorithm that will inherit the physical properties of the underlying continuum problem. A dissipative algorithm not only simulates the actual physical process, but also guarantees stability of the algorithm in the sense of energy control (see ARMERO & PETŐCZ[4]).

### 6.2 Dissipative friction algorithm

As stated above, friction is a dissipative phenomenon so it is of interest to study the evolution of the energy of the system in the presence of frictional contact forces.

To investigate the contribution of the tangential contact forces to the variation of the total

energy  $\mathcal{E}_t$ , we state the variation of the energy for a typical interval of time  $[t_n, t_{n+1}]$  as in equation 5.1, where we have used  $\varphi_{n+1} - \varphi_n$  as our admissible variation  $\delta\varphi$  in the weak form equation 3.1. By performing this substitution in the expression for  $\delta\xi^\beta$  in equation 3.4, the energy variation now takes the form

$$\begin{aligned} \mathcal{E}_{n+1} - \mathcal{E}_n &= \int_{\cup \Gamma_c} t_N (g_{n+1}^d - g_n^d) d\Gamma \\ &\quad - \int_{\cup \Gamma_c} t_{T\alpha} A_{\alpha\beta}^{n+\frac{1}{2}-1} \left[ \llbracket \varphi_{n+1} - \varphi_n \rrbracket \cdot \tau_\beta^{(n+\frac{1}{2})} + g_{n+\frac{1}{2}} (\varphi_{n+1,\beta}^2 - \varphi_{n,\beta}^2) \cdot \nu_{n+\frac{1}{2}} \right] d\Gamma \end{aligned} \quad (6.1)$$

Analogous to the definition of the dynamic gap function  $g^d$ , let us define the *dynamic slip*, denoted by  $\xi^d$ , in the following way [3]:

$$A_{\alpha\beta}^{n+\frac{1}{2}} (\xi_{n+1}^{d,\beta} - \xi_n^{d,\beta}) := \llbracket \varphi_{n+1} - \varphi_n \rrbracket \cdot \tau_\beta^{(n+\frac{1}{2})} + g_{n+\frac{1}{2}} (\varphi_{n+1,\beta}^2 - \varphi_{n,\beta}^2) \cdot \nu_{n+\frac{1}{2}}, \quad (6.2)$$

where the geometrical quantity  $A_{\alpha\beta}^{n+\frac{1}{2}}$  is defined in equation 2.44 and is evaluated at the configuration at  $t_{n+\frac{1}{2}}$ . Also, the definition of the normal  $\nu_{n+\frac{1}{2}}$  is obtained by means of a closest point projection performed at  $t_{n+\frac{1}{2}}$ .

**Remark 6.1** Equation 6.2 is a second order approximation of the evolution equation for the tangential slip  $\xi^\alpha$  expressed in equation 2.43.

Substituting definition 6.2 into equation 6.1, the expression for the energy variation becomes

$$\mathcal{E}_{n+1} - \mathcal{E}_n = \int_{\cup \Gamma_c} \left[ t_N (g_{n+1}^d - g_n^d) - t_{T\alpha} (\xi_{n+1}^{d,\alpha} - \xi_n^{d,\alpha}) \right] d\Gamma. \quad (6.3)$$

**Remark 6.2** Notice that the definition of the dynamic slip in equation 6.2 involves the deformation  $\varphi$  and not the material velocities  $\mathbf{V}$ , so that this contact quantity can be used, with only slight modifications, in a quasi-static problem.



### 6.2.1 Split operator integration algorithm

Recall the equations that govern the friction phenomenon, namely,

$$\phi := \|t_T\| - \mu t_N \leq 0, \quad (6.4)$$

$$\gamma \geq 0, \quad (6.5)$$

$$\gamma \phi = 0, \quad (6.6)$$

$$\gamma \dot{\phi} = 0, \quad (6.7)$$

and the regularized slip rate relation, given by

$$v_T = \gamma \frac{t_T}{\|t_T\|} + \frac{1}{\kappa_T} \dot{t}_{T\alpha} \tau^\alpha. \quad (6.8)$$

On the other hand, contact kinematics yields the following expression for the tangential slip rate  $v_T$ :

$$v_T = M_{\alpha\beta} \dot{\xi}^\beta \tau^\alpha. \quad (6.9)$$

We substitute the dynamic slip rate  $\xi^{d,\beta}$  for  $\xi^\beta$  in equation 6.9 [3], to obtain the following expression:

$$v_T = M_{\alpha\beta} \dot{\xi}^{d,\beta} \tau^\alpha. \quad (6.10)$$

Next, the slip relation 6.8 combined with equation 6.10 is discretized in time by means of a generalized mid-point approximation of the form

$$\bar{M}_{\alpha\beta} (\xi_{n+\theta}^{d,\beta} - \xi_n^{d,\beta}) = \theta \Delta \gamma \frac{t_{Tn+\theta}}{\|t_{Tn+\theta}\|} + \frac{1}{\kappa_T} (t_{Tn+\theta} - t_{Tn}), \quad (6.11)$$

for any  $\theta \in (0, 1]$ . We have used an approximation to the reference metric, denoted by  $\bar{M}_{\alpha\beta}$ ; the nature of this approximation is explained below.

The slip surface function  $\phi$  and Kuhn-Tucker conditions for a typical time interval  $[t_n, t_{n+1}]$  now take the following form:

$$\phi = \|t_{Tn+\theta}\| - \mu t_N \leq 0, \quad (6.12)$$

$$\Delta\gamma \geq 0, \quad (6.13)$$

$$\Delta\gamma \phi = 0. \quad (6.14)$$

To integrate equation 6.11, while enforcing the Kuhn-Tucker complementary conditions 6.13 and 6.14, a operator split strategy is used:

### 1. Trial state

As in the standard return mapping algorithm, we first evaluate the trial state by assuming a stick phase by setting  $\Delta\gamma = 0$ , thus obtaining an expression for the trial state tangent stress  $t_{Tn+\theta}^{trial}$  at  $t_{n+\theta}$ , that is,

$$t_{Tn+\theta}^{trial} = \kappa_T \bar{M}_{\alpha\beta} \left( \xi_{n+\theta}^{d,\beta} - \bar{\xi}^{d,\beta} \right), \quad (6.15)$$

where we have assumed that stick conditions apply for  $N$  time increments. In other words,  $\Delta\gamma = 0$  for  $n = 0, \dots, N-1$  and the initial point of contact at the initiation of the  $N$  stick time steps is  $\bar{\xi} = \bar{\xi}^d$ .

To decide which of the frictional states, slip or stick, is present, we evaluate the function  $\phi$  using this trial stress. We denote this value  $\phi^{trial}$ .

### 2. Stick phase

If  $\phi^{trial} < 0$ , then  $\Delta\gamma = 0$  must hold to satisfy the condition 6.14. Consequently, we satisfy the stick condition and the trial tangent stress in equation 6.15 is the actual frictional stress, i.e.,

$$\begin{aligned} t_{Tn+\theta} &= t_{Tn+\theta}^{trial} \\ &= \kappa_T \bar{M}_{\alpha\beta} \left( \xi_{n+\theta}^{d,\beta} - \bar{\xi}^{d,\beta} \right), \end{aligned} \quad (6.16)$$

### 3. Slip phase

Assume that step  $N + 1$  is a slip step, that is

$$\phi^{trial} \geq 0, \quad (6.17)$$

meaning that we need to correct the trial frictional stress  $t_{Tn+\theta}^{trial}$  so that it satisfies condition 6.14. Since a slip state implies  $\Delta\gamma > 0$ , we know that

$$\phi = 0. \quad (6.18)$$

Therefore, substituting the expression of the trial state frictional stress in equation 6.15 into equation 6.14 yields

$$t_{TN+\theta}^{trial} = t_{TN+\theta} \left( 1 + \frac{\kappa_T \Delta\gamma}{\|t_{TN+\theta}\|} \right), \quad (6.19)$$

implying that the trial and actual stress states are parallel, i.e.

$$\frac{t_{TN+\theta}^{trial}}{\|t_{TN+\theta}^{trial}\|} = \frac{t_{TN+\theta}}{\|t_{TN+\theta}\|}. \quad (6.20)$$

This last result is crucial to solve the problem in closed form. Using equation 6.20 together with condition 6.18, we obtain

$$t_{TN+\theta} = \mu t_N \frac{t_{TN+\theta}^{trial}}{\|t_{TN+\theta}^{trial}\|}, \quad (6.21)$$

with  $\Delta\gamma = \frac{\phi^{trial}}{\theta \kappa_T}$ .

A slip frictional state involves relative slip between the two surfaces and we need to update our stick point to the new contact point. That is,  $\bar{\xi}^d = \xi_{N+1}^d$ .

**Remark 6.3** The proposed frictional scheme uses a constant reference metric  $\bar{M}_{\alpha\beta}$  evaluated at the stick point  $\bar{\xi}^d$  for simplicity. As  $\kappa_T \gg 0$ , the slip is very small during a stick step, so this simplification does not add any significant limitation to the frictional scheme.

### 6.3 Energy evolution of the frictional algorithm

In this section we prove that our scheme is unconditionally dissipative. Including the results from the previous chapter for the energy evolution in the presence of normal contact forces, we have

$$(\mathcal{E}_{n+1} + \mathcal{P}_{n+1}) - (\mathcal{E}_n + \mathcal{P}_n) = - \int_{\cup \Gamma_c} t_{T\alpha} \left( \xi_{n+1}^{d,\alpha} - \xi_n^{d,\alpha} \right) d\Gamma. \quad (6.22)$$

For each frictional state we have a different expression for the tangent friction force. We treat each one individually:

#### 1. Stick phase

As in the previous section, assume that there are  $N$  consecutive steps which satisfy the stick condition. For later use, we define the following notation:

$$\eta_t^d := \xi_t^d - \bar{\xi}^d. \quad (6.23)$$

Also, we define a norm by means of the approximate metric  $\bar{M}_{\alpha\beta}$ :

$$\|\eta\|_{\bar{M}} = \eta^\alpha \bar{M}_{\alpha\beta} \eta^\beta. \quad (6.24)$$

Using expression 6.16, the change in total energy for a typical time step  $[t_n, t_{n+1}]$  is as follows:

$$\begin{aligned}
 (\mathcal{E}_{n+1} + \mathcal{P}_{n+1}) - (\mathcal{E}_n + \mathcal{P}_n) &= - \int_{\cup \Gamma_c} \kappa_T \left( \xi_{n+1}^{d,\alpha} - \xi_n^{d,\alpha} \right) \bar{M}_{\alpha\beta} \left( \xi_{n+\theta}^{d,\beta} - \bar{\xi}^{d,\beta} \right) d\Gamma \\
 &= - \int_{\cup \Gamma_c} \kappa_T \left( (\xi_{n+1}^{d,\alpha} - \bar{\xi}^{d,\alpha}) - (\xi_n^{d,\alpha} - \bar{\xi}^{d,\alpha}) \right) \bar{M}_{\alpha\beta} \\
 &\quad \left( \theta(\xi_{n+1}^{d,\beta} - \bar{\xi}^{d,\beta}) - (1-\theta)(\xi_n^{d,\beta} - \bar{\xi}^{d,\beta}) \right) d\Gamma \\
 &= - \int_{\cup \Gamma_c} \kappa_T (\eta_{n+1}^{d,\alpha} - \eta_n^{d,\alpha}) \bar{M}_{\alpha\beta} \\
 &\quad \left( \frac{(\eta_{n+1}^{d,\beta} + \eta_n^{d,\beta})}{2} + \left( \theta - \frac{1}{2} \right) (\eta_{n+1}^{d,\beta} + \eta_n^{d,\beta}) \right) d\Gamma \\
 &= - \int_{\cup \Gamma_c} \left[ \kappa_T \left( \frac{\|\eta_{n+1}^d\|_{\bar{M}}^2}{2} - \frac{\|\eta_n^d\|_{\bar{M}}^2}{2} \right) \right. \\
 &\quad \left. - \kappa_T \left( \theta - \frac{1}{2} \right) \|\eta_{n+1}^d - \eta_n^d\|_{\bar{M}}^2 \right] d\Gamma. \tag{6.25}
 \end{aligned}$$

The variation of the total energy after  $N$  steps is then

$$\begin{aligned}
 \sum_{n=0}^{N-1} (\mathcal{E}_{n+1} + \mathcal{P}_{n+1}) - (\mathcal{E}_n + \mathcal{P}_n) &= - \int_{\cup \Gamma_c} \kappa_T \left( \frac{\|\eta_N^d\|_{\bar{M}}^2}{2} - \frac{\|\eta_0^d\|_{\bar{M}}^2}{2} \right) d\Gamma \\
 &\quad - \left( \theta - \frac{1}{2} \right) \sum_{n=0}^{N-1} \int_{\cup \Gamma_c} \kappa_T \|\eta_{n+1}^d - \eta_n^d\|_{\bar{M}}^2 d\Gamma \\
 &\leq 0, \tag{6.26}
 \end{aligned}$$

where  $\eta_0^d = 0$ , since the stick point  $\bar{\xi}^d$  was initialized as  $\xi_0^d$ . This observation, together with the fact that  $\theta \geq \frac{1}{2}$ , allows us to establish that the two integrands in 6.26 are positive.

**Remark 6.4** It is important to notice that even though the scheme introduces some dissipation in the stick phase (due to the approximate way in which we satisfy the stick constraint), by choosing  $\theta = \frac{1}{2}$ , we can minimize this artificial dissipation. Also, notice that in the limit as  $\kappa_T \rightarrow \infty$ , this dissipation vanishes.

## 2. Slip phase

To calculate the variation of the energy for a slip step, we use expression 6.21 to substitute into equation 6.22, and we obtain

$$\begin{aligned}
 (\mathcal{E}_{n+1} + \mathcal{P}_{n+1}) - (\mathcal{E}_n + \mathcal{P}_n) &= - \int_{\cup \Gamma_c} \mu t_N \frac{t_N^{trial}}{\|t_{Tn+\theta}^{trial}\|} (\xi_{n+1}^{d,\alpha} - \xi_n^{d,\alpha}) d\Gamma \\
 &= - \int_{\cup \Gamma_c} \frac{\kappa_T \mu t_N}{\|t_{Tn+\theta}^{trial}\|} ((\xi_{n+1}^{d,\alpha} - \bar{\xi}^{d,\alpha}) - (\xi_n^{d,\alpha} - \bar{\xi}^{d,\alpha})) \\
 &\quad \bar{M}_{\alpha\beta} (\theta(\xi_{n+1}^{d,\beta} - \bar{\xi}^{d,\beta}) - (1-\theta)(\xi_n^{d,\beta} - \bar{\xi}^{d,\beta})) d\Gamma \\
 &= - \int_{\cup \Gamma_c} \frac{\kappa_T \mu t_N}{\|t_{Tn+\theta}^{trial}\|} (\eta_{n+1}^{d,\alpha} - \eta_n^{d,\alpha}) \bar{M}_{\alpha\beta} \\
 &\quad \left( \frac{(\eta_{n+1}^{d,\beta} + \eta_n^{d,\beta})}{2} + \left(\theta - \frac{1}{2}\right)(\eta_{n+1}^{d,\beta} + \eta_n^{d,\beta}) \right) d\Gamma \\
 &= - \int_{\cup \Gamma_c} \left[ \frac{\kappa_T \mu t_N}{\|t_{Tn+\theta}^{trial}\|} \left( \frac{\|\eta_{n+1}^d\|_{\bar{M}}^2}{2} - \frac{\|\eta_n^d\|_{\bar{M}}^2}{2} \right) \right. \\
 &\quad \left. - \frac{\kappa_T \mu t_N}{\|t_{Tn+\theta}^{trial}\|} \left(\theta - \frac{1}{2}\right) \|\eta_{n+1}^d - \eta_n^d\|^2 \right] d\Gamma. \quad (6.27)
 \end{aligned}$$

If we now consider the contributions of the  $N$  stick time steps followed by a final slip time step, we obtain the following expression for the variation of the energy for the time interval  $[0, t_N]$ :

$$(\mathcal{E}_N + \mathcal{P}_N) - (\mathcal{E}_0 + \mathcal{P}_0) \leq - \int_{\cup \Gamma_c} \left[ \frac{\kappa_T \mu t_N}{\|t_{Tn+\theta}^{trial}\|} \frac{\|\eta_{N+1}^d\|_{\bar{M}}^2}{2} - \kappa_T \left(1 - \frac{\mu t_N}{\|t_{Tn+\theta}^{trial}\|}\right) \frac{\|\eta_N^d\|_{\bar{M}}^2}{2} \right] d\Gamma \quad (6.28)$$

where we have discarded the  $\|\eta_{n+1} - \eta_n\|$  terms to arrive at the inequality. The first term of the integrand is positive due to the fact that normal contact involves compressive tractions on the contact surface, that is,  $t_N \geq 0$ ; and the expression in parenthesis of the second term is positive due to the inequality 6.17 which holds for the slip time step.

Thus, the energy is dissipated throughout frictional contact conditions; this contributes to the numerical stability of the algorithm. It can be shown that the conventional frictional schemes cannot be proven to dissipate unconditionally if the dynamical slip  $\xi^d$  is not introduced into the scheme; this not only makes the algorithm energetically unstable but also represents an infeasible physical situation.

## 6.4 Space discretization

We now develop the expressions for the frictional contact problem, in a finite element formulation in the two dimensional setting.

The tangent traction is expressed in a space discretized setting. For each slave node  $S$  with coordinates  $\mathbf{X}_S$  we can define the variation  $\delta\xi^\alpha$  in the following way:

$$\xi_{s,n+1}^{d,\alpha} = \xi_{s,n}^{d,\alpha} + \underbrace{\frac{1}{m_{11} - g_{s,n+\frac{1}{2}}\nu_{n+\frac{1}{2}}e_{1,1}(\bar{\xi}^1)} \left[ \hat{T}_{n+\frac{1}{2}}^T - \frac{g_{s,n+\frac{1}{2}}}{l_s} \hat{D}_{n+\frac{1}{2}}^T \right]}_{=: \hat{H}_{n+\frac{1}{2}}} (\hat{d}_{n+1} - \hat{d}_n),$$

with the vectors  $\hat{D}_{n+\frac{1}{2}}$  and  $\hat{T}_{n+\frac{1}{2}}$  defined as:

$$\hat{D}_{n+\frac{1}{2}} = \begin{bmatrix} 0 \\ -N_{,\xi}^{M1}(\xi_c)\nu_{n+\frac{1}{2}} \\ -N_{,\xi}^{M2}(\xi_c)\nu_{n+\frac{1}{2}} \\ \vdots \end{bmatrix} \quad \text{and} \quad \hat{T}_{n+\frac{1}{2}} = \begin{bmatrix} \tau_{n+\frac{1}{2}} \\ -N^{M1}(\xi_c)\tau_{n+\frac{1}{2}} \\ -N^{M2}(\xi_c)\tau_{n+\frac{1}{2}} \\ \vdots \end{bmatrix}. \quad (6.29)$$

We have used the following compact notation to express the displacements of the slave and master segment nodes  $\{S, M1, M2, \dots\}$ :

$$\hat{d}_{s,n+\frac{1}{2}} = \begin{bmatrix} d_{n+\frac{1}{2}}^s \\ d_{n+\frac{1}{2}}^{M1} \\ d_{n+\frac{1}{2}}^{M2} \\ \vdots \end{bmatrix}. \quad (6.30)$$

We assume that friction is present in the problem as long as the bodies are in contact, so friction will be added to the problem following the same contact/release criteria of the frictionless problem (see Chapter 5).

Using the previous notation, the total contact force  $\hat{f}_c^{(n+\frac{1}{2})}$  in equation 5.20 has the following form:

$$\hat{\mathbf{f}}_{s,c}^{(n+\frac{1}{2})} = t_N \hat{\mathbf{G}}_{s,n+\frac{1}{2}} + t_{Tn+\theta} \hat{\mathbf{H}}_{s,n+\frac{1}{2}}. \quad (6.31)$$

Following standard procedure in the finite element formulation, the force contribution of each contact element is assembled as in equation 4.36. The expression for the frictional traction  $t_T$  for a particular contact element  $\{S, M1, M2, \dots\}$  for a typical time interval  $[t_n, t_{n+1}]$  takes the following form:

$$t_{Tn+\theta\alpha} = \begin{cases} t_{Tn+\theta\alpha}^{trial} & \text{if } \phi \leq 0 \\ \mu t_N \frac{t_{Tn+\theta\alpha}^{trial}}{\|t_{Tn+\theta\alpha}^{trial}\|} & \text{if } \phi > 0 \end{cases} \quad (6.32)$$

where

$$t_{Tn+\theta\alpha}^{trial} = \kappa_T \bar{M}_{\alpha\beta} \left[ \theta \xi_{s,n+1}^{d,\beta} + (1 - \theta) \xi_{s,n}^{d,\beta} - \bar{\xi}_S^\beta \right]. \quad (6.33)$$

The stick point  $\bar{\xi}_s$  takes its value from the converged value of the slip from the last time step for which the slip condition was detected (i.e. for which  $\phi^{trial} \geq 0$ ).



# Chapter 7

## Sorting Algorithm

### 7.1 Introduction

One of the most important parts of a good contact/impact algorithm is the contact detection scheme, in terms of its speed and efficiency. The contact detection becomes a cumbersome task as the system of bodies becomes large. This task would include performing  $N^2$  closest point projections among  $N$  bodies. So the need arises for a more efficient way of discovering the geometric relationships among objects within a working space, because this process accounts for a significant portion of the computational effort of solving contact/impact problems among many bodies.

Contact detection can be defined as finding the members of a set of points that lie inside a subregion of an  $N$  dimensional space. In our formulation, the set of points is the set of nodes which lie on the surface of each body, and each body defines a subregion in a two dimensional space.

In a multi-body system, it makes sense to distinguish two phases in the contact detection procedure: a spatial sorting phase and a contact resolution phase (see WILLIAMS & O'CONNOR[35]). The spatial sorting enables us to find all the pairs of bodies which could be potential contactors and the contact resolution phase finds the actual two points on the surface of each pair of bodies.

In this chapter we expand on the issues involved in the spatial sorting phase of the contact detection algorithm.

## 7.2 Overview of various sorting techniques

In general, it is necessary to consider a system of bodies which evolves with time. The efficiency of the contact detection algorithm can be greatly improved if one can assume *a priori* knowledge of how this system will evolve.

Algorithms assuming *a priori* knowledge can be extremely efficient but have the disadvantage that the range of problems they can tackle is limited. The following is a brief overview of the kind of *a priori* knowledge that can be used to produce a fast detection algorithm taken from WILLIAMS & O'CONNOR[35]:

### 1. Fixed topology

Fixed topology can be found in finite element algorithms when the relative position of the elements remains unchanged during the simulation of a particular problem.

### 2. Slowly varying topology

The objects move around only a small amount according to some metric, so that each object only interacts with its neighbours and we only keep track of a small number of possible contactors per body. If one can keep track of the characteristic velocities in a problem, then it is also possible to check for contact after only a certain number of time steps instead of after every one.

### 3. Spatially sparse systems

If the system is very sparse, it makes sense to project trajectories in such a way that we only check for the intersection of cones or cylinders in a space-time system. Animation applications have used this scheme successfully, (see e.g. DWORKIN[9]).

### 4. Exhaustive spatial sorting schemes

If the scheme makes no *a priori* assumptions about the evolution of the problem and

it is based only on the present geometric configuration, it is called *exhaustive*. These schemes are more general and robust but are potentially slower than non-exhaustive schemes.

## 5. Spatial sorting algorithms

Spatial sorting gives us a valuable tool to decide which bodies should be considered for a more detailed contact resolution. It seeks to avoid the all-to-all body search for contact at each time step. For a small quantity of objects, this all-to-all method is acceptable. (Shi considers a maximum of 64, in [28].) But, for example, a simulation involving exhaustive searching over a thousand objects can become computationally prohibitive.

In this work we use a *spatial sorting technique* combined with the traditional closest point projection as our contact resolution method. Below, we review several of the most powerful methods as classified by WILLIAMS & O'CONNOR[35] and in subsequent sections we describe in some detail the *binary space partitioning* which has been used in the present work.

### 1. Grid subdivision

The grid subdivision method discretizes in a uniform way the simulation volume or area into rectilinear cells, each cell enclosing one or more objects. Objects are associated with each cell based on their coordinates and neighbouring objects are detected by their cell assignment. The performance of this method depends greatly on the homogeneity of the spatial distribution of the objects within the working space and is not a good overall methodology for a wide range of problems.

### 2. Adaptive grid methods

Adaptive cell methods are used to avoid the problems associated with simple *grid subdivision*, though at the cost of managing multiple cell dimensions. With this approach, the scheme suffers when the distribution of objects becomes homogeneous.

### 3. Body based cells

Another alternative is to base the cell surrounding an object on its centroid. Any

object lying within this cell is considered to be a potential contactor and added to a *neighbour list*. The multi-pole algorithm of GREENGARD & ROHKLIN [11] makes use of a similar approach in the area of molecular dynamics.

#### 4. Spatial heap-sort

Heap-sort is one of several algorithms used to sort the objects into an ordered list which at the same time gives an indication of the location of each particular object. In this case, the key to the ordered list is the object's coordinate along one or more global axes. This method has been applied to particle hydrodynamics by SWEGLER [32]. A modified heap-sort algorithm called DFR has been developed by WILLIAMS & O'CONNOR [35] and applied to baseline granular simulation problems.

#### 5. Tree methods

Binary sort/search algorithms provide a flexible and general methodology for contact detection in two dimensional problems. The octree sort/search algorithm was derived from the binary one to handle problems in three dimensions. The technique consists of treating the objects as rectilinear cells; however in this case the cells are ordered into a tree data structure. To ensure the creation of a balanced tree, Knuth [17] suggests that insertion into the tree should be done in a relatively random fashion. The time required to create a binary tree is of the order  $\mathcal{O}(N \log(N))$ , where  $N$  is the total number of objects in the problem; idem for the traversing of the tree. In short, such an algorithm is a valuable improvement over an all-to-allsearch, which is  $\mathcal{O}(N^2)$ .

In our work, we implement a binary tree database structure to perform the sorting (see e.g. MUNJIZA ET AL[24] and BONET & PERAIRE[6]). It has proven to be a versatile tool that can be applied to a wide range of problems in the area of contact/impact simulations.

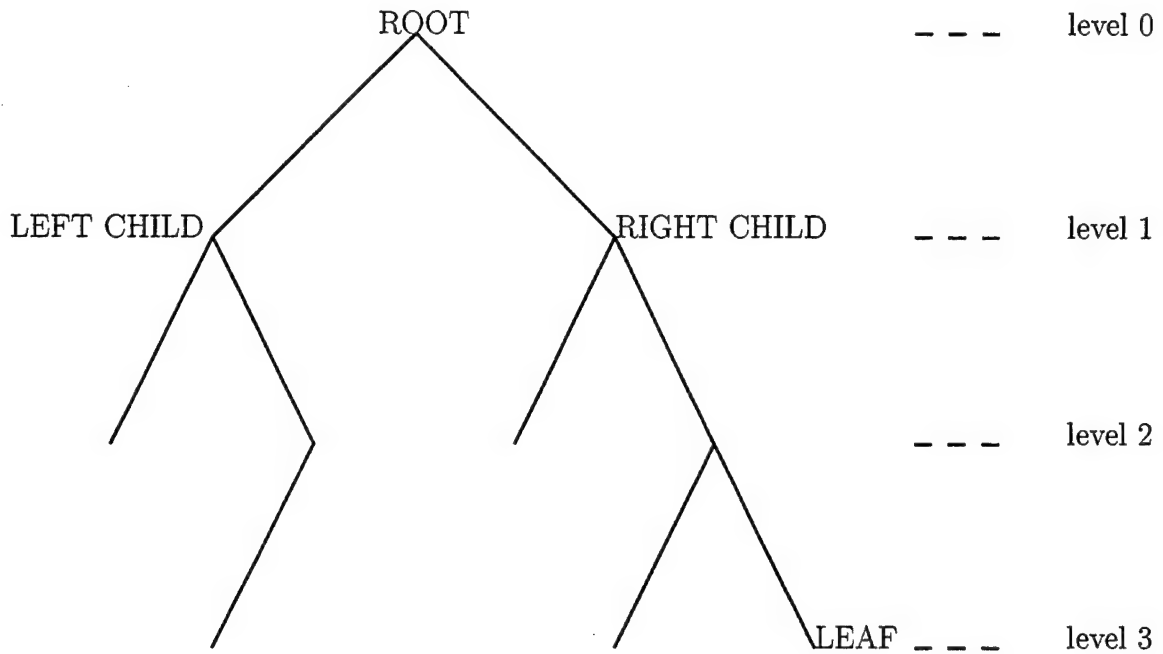


Figure 7.1: Schematic drawing of a binary tree

## 7.3 Binary space partitioning

### 7.3.1 Binary tree structure

Originally, tree structures were developed to store in a systematic way a collection of data in order to enable quick access. A tree consists of *nodes* where the actual data is stored. Each node is extended by the addition of two links to two other nodes known as the *left child* and the *right child*. The node from which a particular node springs is called the *parent* node. Each tree has a starting node which we name *root*. Also, at each node, there is a *subtree* originating from it so that the node becomes the root of this subtree.

This definition establishes a hierarchy of nodes: the root at the top level; 0, 1 or 2 nodes at the next level, each of which in turn has 0, 1 or 2 nodes at the next level of hierarchy; and so forth. A node without children is said to be a *leaf*. This hierarchical structure inspires the graphical representation shown in Figure 7.1.

### 7.3.2 Space partitioning using a binary tree

The first step in a sorting algorithm is to construct the binary tree. Consider a problem with  $N$  bodies within an area which is called the *searching space* or *working space*. In the definition of the *searching space*, we use the following *a priori* assumption: during the time interval of interest, the bodies in question never leave the searching space, so its area remains unchanged throughout the evolution of the problem.

Let us consider the set of surface nodes corresponding to the surfaces of the  $N$  bodies and denote by  $n_{total}$  the size of this set of nodes.

**Remark 7.1** From the point of view of the previous chapters, this set of nodes corresponds to the surface nodes of the slave and master bodies which participate in the contact problem prior to any closest point projection procedure. The set of surface nodes of the master body must not be confused with the master segment identified by the closest point projection.

**NOTE:** In the rest of the chapter, we give the name *particles* to the set of  $n_{total}$  nodes belonging to the surfaces of the bodies participating in the contact problem, and we give the name *nodes* to the nodes of the binary tree using in storing our data.

The root of the binary tree is associated with our chosen initial searching space. Let us consider a particle with its corresponding current coordinates. We divide the working space in half in a particular direction, say vertically, and then determine on which side this particle falls; then, we create a child which we associate with that half space. We think of this procedure as *inserting* the particle in a node. Next, we consider another particle. If this new particle falls on the same half space as the previous one, we subdivide the half space and we move the previously inserted particle into the corresponding quarter space, and insert the new particle into the quarter space where it belongs. The procedure is repeated until each particle resides on a leaf. On the other hand, if the new particle falls on the opposite half space as the previous one, we just create the second child in the tree and continue the process by considering the next particle in the queue.

**Remark 7.2** Notice that we do not create a child node in our tree unless there is a particle

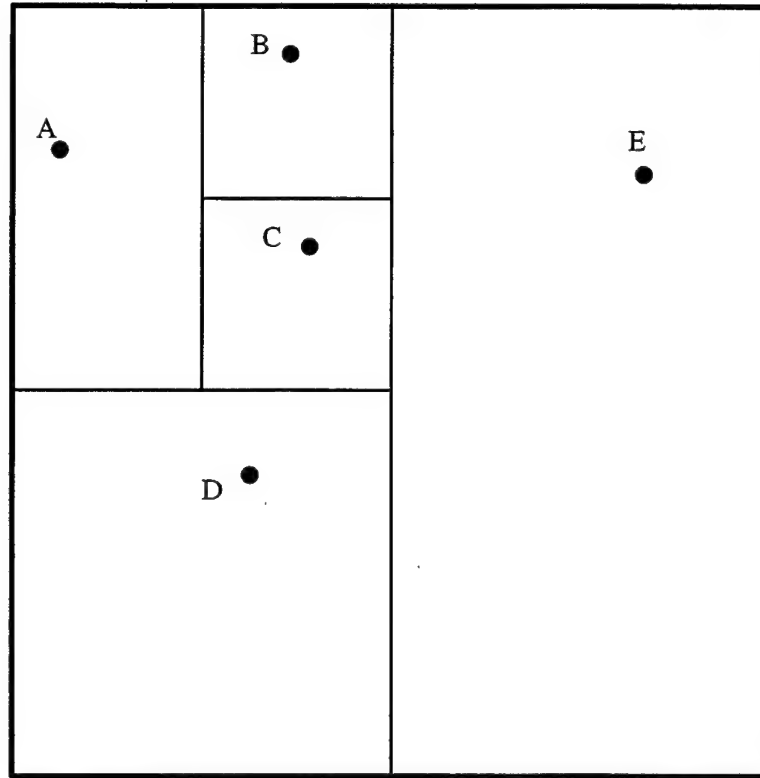


Figure 7.2: Schematic drawing of a set of particles within a square working space

occupying it. This is a modified way of building a tree and avoids the creation of nodes which later would have to be deleted because they are empty. Other authors use different versions of tree building algorithms which are more suitable for their purposes (see e.g. BONET & PERAIRE[6]).

When we finish inserting all the particles in the tree, they all reside on leaves, i.e. there is a unique particle residing in each cell.

#### 7.3.2.1 Construction of the binary tree example

In Figures 7.2 and 7.3 we show an example of a set of particles which have been inserted into a tree.

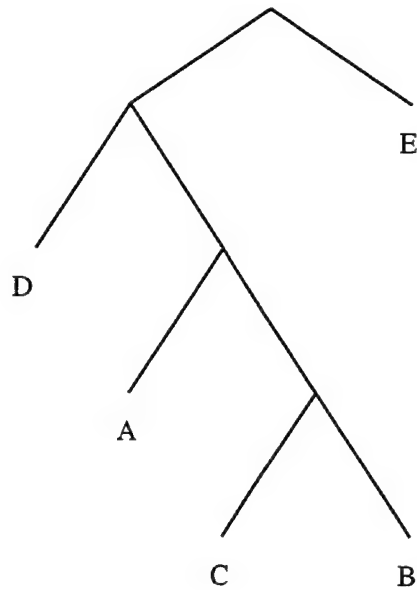


Figure 7.3: Binary tree structure corresponding to the example shown in Figure 7.2

#### 7.3.2.2 Binary tree construction program

Given a list of particles coordinates, the binary tree can be constructed recursively as follows:

##### Procedure BintreeBuild

```

Bintree = {empty}
do i = 1:ntotal
  Call BinInsert(i,root)
end do

```

##### Procedure BinInsert(i,n)...insert particle i in node n

*if the subtree rooted at n contains more than 1 particle*

Determine which child c of node n particle i lies in

Call BinInsert(i,c)

*else if the subtree rooted at n contains 1 particle, n is a leaf*

Consider n's 2 children, create the child in which

the particle already in n lies and move particle i into it.

Let c be the child in which particle i lies



```
    Call BinInsert(i,c)
else if the subtree rooted at n is empty, n is a leaf
    Store particle i in node n
end if
```

## 7.4 Sorting using binary tree

In this section, we explain how best to use the data stored in the binary tree in a general sorting algorithm (see MUNJIZA ET AL[24]). Once the binary tree has been constructed, as described in the previous sections, we proceed to the actual sorting phase of the algorithm. We search for possible contact pairs between any of the  $n_{total}$  particles and the  $N$  bodies. Recall that the particles are the actual surface nodes (in the finite element sense) of these bodies.

**Remark 7.3** We assume that a node cannot penetrate the body it belongs to, thus eliminating self contact cases from the algorithm.

To check for contact at this stage, we construct a *buffer zone* around each body to simplify the search for contact [24]. An example of a buffer zone is the circumscribing rectangle which is aligned with the reference coordinate system (see Figure 7.4). The coordinates of this rectangle are the minimum and maximum coordinates in each space dimension that bound the object. In our work, we have used this type of buffer zone, which does not assume any *a priori* knowledge of the problem.

**Remark 7.4** The calculation of the buffer zone is related to the sorting scheme, since it depends on how often one performs a sorting procedure. If a sorting procedure is performed every  $n$  time steps, the buffer zone is chosen so that its thickness  $b$  is at least  $b > nv_{max}\Delta t$ , where  $v_{max}$  is the maximum velocity among all the bodies.

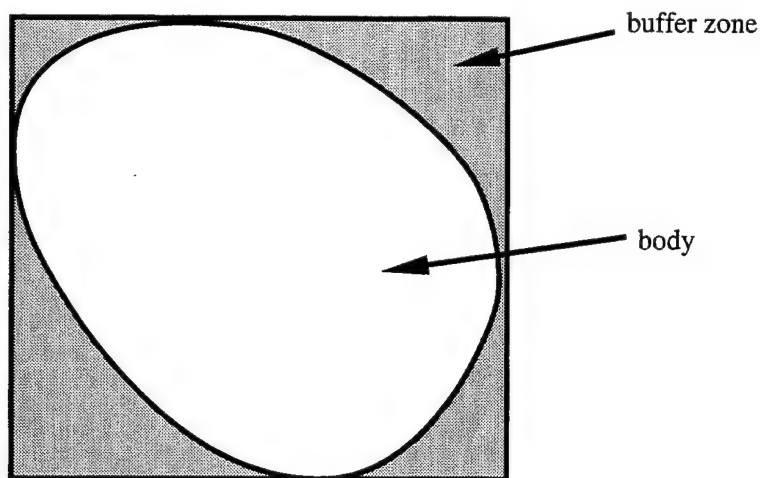


Figure 7.4: Schematic drawing of a body and its corresponding buffer zone

### 7.4.1 Traversing the binary tree

The contact search is performed by *traversing* the tree. The binary tree is traversed  $N$  times (once for each body). We *visit* each node on the tree, and it has associated with it a square or rectangular *cell*. The target body with its designated buffer zone is checked for superposition with this cell. Efficiency is improved if we take into account that if no superposition with a node is detected, then no further contact detection need be performed for the corresponding subtree rooted at that node.

When we reach a *leaf* of the tree which contains an actual particle (slave or master node), the particle is checked for inclusion within the body in question.

To traverse the tree we use the recursive *preorder scheme*. (See [17] for the basics on traversing trees and preorder schemes.)

#### 7.4.1.1 Preorder scheme

The pseudo code is presented below

```
subroutine preorder(current-address)
```

```
call visit (Visit node stored at current-address)
if (cont = .true.) then
  if (current-left-child exists) then
    call preorder(current-left-child-address)
  end if
  if (current-right-child exists) then
    call preorder(current-right-child-address)
  end if
end if
stop
end
```

*Visiting* a node means checking for superposition of two rectangles (the subpartition associated with the tree node and the buffer zone rectangle) or checking whether a particle lies within a rectangle (the finite element node and the buffer zone rectangle). During the “visit”, then, we recognize a *pair* consisting of a particle and a body as being in possible contact. This pair, which we call *bp-pair* for **body-particle-pair**, is then added to the list of possible pairs in contact.

**Remark 7.5** Even though we do not assign *slave* and *master* categories to the particles in our problem before the sorting phase, the categorization comes up automatically. The closest point projection phase of the algorithm will consider that the particle in the bp-pair is the *slave node* to be checked for penetration into the body belonging to the bp-pair.

## 7.5 Implementation of a sorting algorithm

The type of database involved in the sort/search procedure requires the use of data structures and pointers. FORTRAN 90 provides us with these necessary tools.

The design and definition of our database structure are of importance, as they affect the efficiency and memory use of the algorithm. Below, we present some of the most important

structures and issues involved in the implementation.

### 7.5.1 Binary tree

A node in the tree is defined as follows:

```
type node
  type(particle),pointer :: part-in-node
  real(8), dimension(2) :: xmin,xmax
  type(node),pointer :: parent,left-child,right-child
end type node
```

Pointers are used to point at other pieces of data, thus eliminating the need to make multiple copies of the data. For example, when traversing the binary tree, we need to go down to the children of a certain node, so we include in the node data structure two pointers, one to each child.

The first part of the **node** data structure is the *part-in-node*, which is a pointer to a **particle** type structure. If the node is not a leaf then this pointer is *nullified* (does not point to any data). When this happens, the pointer is said to be *disassociated*.

### 7.5.2 Body, surface and particle

A body is formed by a set of surfaces and each one of these surfaces consists of a set of particles. Following this reasoning, we can think of a particle as being a substructure of a surface and a surface as a substructure of a body.

Keeping this in mind, we define the following structures:

```
type body
  integer :: num-surf
  type(surface),dimension(:),pointer :: id-surf
```

```
    real(8), dimension(2) :: lim-min,lim-max  
end type body
```

The integer *num-surf* indicates the number of surfaces that define the body. The real numbers *lim-min* and *lim-max* contain the data of the buffer zone rectangle. The pointer array *id-surf* points to each surface data structure, as described.

**Remark 7.6** The capability of defining a body by means of multiple surfaces, enables the scheme to simulate a body with different surface properties. For example, a body may have different surface finishes, each with a different friction coefficient.

```
type surface  
    integer :: num-part  
    type(surface),dimension(:),pointer :: id-part  
end type surface
```

The integer *num-part* indicates the number of particles that define the surface. The pointer array *id-part* points to each particle data structure as described.

```
type particle  
    integer :: glob-num,b-id,s-id  
    real(8) :: x,y  
end type particle
```

The integers *glob-num*, *b-id* and *s-id* represent the global node number (in the finite element sense), the body number and surface number which this particle belongs to, respectively.

### 7.5.3 Linked lists

The output of the sorting phase, the set of *bp-pairs*, is stored in a linked list. The data structure *bp-pair* is defined as follows:

```

type bp-pair
  type(body),pointer :: bp-pair-body
  type(particle),pointer :: bp-pair-part
  type(bp-history),pointer :: bp-pair-hist1, bp-pair-hist2
end type bp-pair

```

The pointer *bp-pair-body* points to the body data that is in contact with the particle whose data is pointed to in turn by *bp-pair-part*. The data *bp-pair-hist1* and *bp-pair-hist2* correspond to two history arrays belonging to data at  $t_n$  and  $t_{n+1}$ , respectively, and have the following data structure:

```

type bp-history
  logical :: cont
  integer,dimension(3) :: ixl
  real(8) :: norm(2), gapd
  ... other data needed for the calculation of the force and tangent matrix contribution
end type bp-history

```

The logical variable *cont* is the contact flag for the considered time step considered. The integer array *ixl* contains the global node numbers (in the finite element sense) of the three nodes that form our three-node contact element. The real variable *gapd* is the dynamic gap which we need to keep track of for our restoring scheme.

When the sorting scheme detects a possible contact, the *bp-pair* is added to a linked list. When no contact is detected by the closest point projection procedure, these *bp-pairs* are taken out from the linked list and the space allocated for them is then deallocated. This procedure has the advantage of using memory in a dynamic way thus saving memory storage.

A linked list is composed of *list-nodes*. The structure of a list-node is as follows:

```

type list-node
  type(bp-pair),pointer :: elem

```

```

type(list-node), pointer :: prev,next
end type list-node

```

The pointer *elem* points to the information of the bp-pair that is contained in the given *list-node* and the pointers *prev* and *next* point to the previous and next list-nodes in the linked list, respectively.

#### 7.5.4 Auxiliary tools

In addition to all the previously defined data structures, we use some auxiliary arrays. One of the most important is an array of pointers called *pairs-table*, of dimension  $n_{total} \times N$ . These pointers are initially nullified, but if a particular bp-pair (with particle number *np* and body number *nb*) , was active at the converged state of the previous time step, then *pairs-table*(*nb*,*np*) points to that particular bp-pair, and the algorithm neither allocates more memory nor generates a second copy of the same bp-pair. On the other hand, if *pairs-table*(*nb*,*np*) is nullified then it is evident that this particular bp-pair was not active at the previous time step, and the algorithm must allocate memory to create it.

#### 7.5.5 Basic algorithm

With the salient points of our database explained, we now detail the basic sorting/search algorithm used in the implementation of our contact/impact scheme.

In the box above, we denote by  $a\%b$  the element *b* of a certain type, where *a* is of that type.

**Example:** Using FORTRAN 90, we may declare

```

type(node) :: a (a is a node in the binary tree) ,

```

so that we have access to the left or right child of *a*, respectively by

$$l = a\%left - child \quad \text{or} \quad r = a\%right - child$$

## BASIC ALGORITHM FOR THE SORT/SEARCH PROCEDURE

```

• Create root
• Update particle coordinate
• Calculate buffer boxes
DO  $i = 1 : n_{total}$ 
    • Insert particle  $i$  in the binary tree
END DO
DO  $i = 1 : N$ 
    • Traverse the tree and check for contact between body  $i$  and
      the particles in each leaf of the tree
END DO
• Delete binary tree
DO WHILE list%bp-pair exists
    IF (list%bp-pair existed at time  $t_n$ (check pairs-table)) THEN
        • Perform closest point projection at  $t_{n+\frac{1}{2}}$  and
          calculate dynamic gap  $g_{n+1}^d$ . Check for contact
    ELSE
        • Perform closest point projection at  $t_n$  and
          calculate real gap  $g_n$ 
        • Perform closest point projection at  $t_{n+\frac{1}{2}}$  and
          calculate dynamic gap  $g_{n+1}^d$  (with  $g_n$ ). Check for contact
    END IF
    IF (cont = .true.) THEN
        • Calculate the force
    END IF
    list  $\Rightarrow$  list%next
END DO

```

Table 7.1: Basic algorithm for the sort/search procedure



where  $l$  and  $r$  are declared as

$$\text{type}(\text{node}) :: l, r \quad (l \text{ and } r \text{ are nodes in the binary tree}),$$

**Remark 7.7** Notice that the binary tree is deleted as soon as we form the linked list containing all the bp-pairs in possible contact.

## 7.6 Computation times

As stated above, the addition of a sorting phase decreases the computational effort involved in the contact detection part of the contact algorithm in multibody problems.

In this section, we measure (1) the average CPU time for sorting/searching in each time step, and (2) the average CPU time for the entire step, for a sample of problem while we vary the number of bodies involved.

These measurements give us an estimate of the speed increase provided by the sorting procedure as the number of bodies increases. It is also of interest to see the fraction of time the contact detection part takes within an overall time step.

Theoretical calculations show that for a homogeneous concentration of particles, which should produce a well-balanced tree, the times for the building and traversing the tree would behave as  $\mathcal{O}(N \log(N))$ , where  $N$  is the number of particles. The detection part of the algorithm with no sorting phase, i.e. an exhaustive search, behaves as  $\mathcal{O}(N^2)$ . In Table 7.2, we show the CPU time in seconds for different problems.

A regression analysis has been performed with the times from Table 7.2 to evaluate the behaviour of each algorithm. With the sorting algorithm, the tested times for detection followed the expected logarithmic dependence on  $N$  with a correlation coefficient of  $R = 0.99985$ , while the times required without sorting matched a quadratic curve with a correlation coefficient of  $R = 0.999991$ . Figures 7.5 and 7.6 show the curves obtained through regression analysis of the data in Table 7.2.

The ratio between the CPU time for the detection phase and the CPU time for the solver

N	Sorting		No sorting	
	$CPU_{\text{detection}}$	$CPU_{\text{total}}$	$CPU_{\text{detection}}$	$CPU_{\text{total}}$
13	0.0085	0.07	0.1244	0.29
53	0.0361	0.90	2.7050	3.60
104	0.0732	1.80	10.8600	12.80
196	0.1386	3.60	40.3800	44.23
400	0.3030	7.64	-	-
900	0.7540	16.60	-	-

Table 7.2: Computing times in seconds for various numbers of bodies.

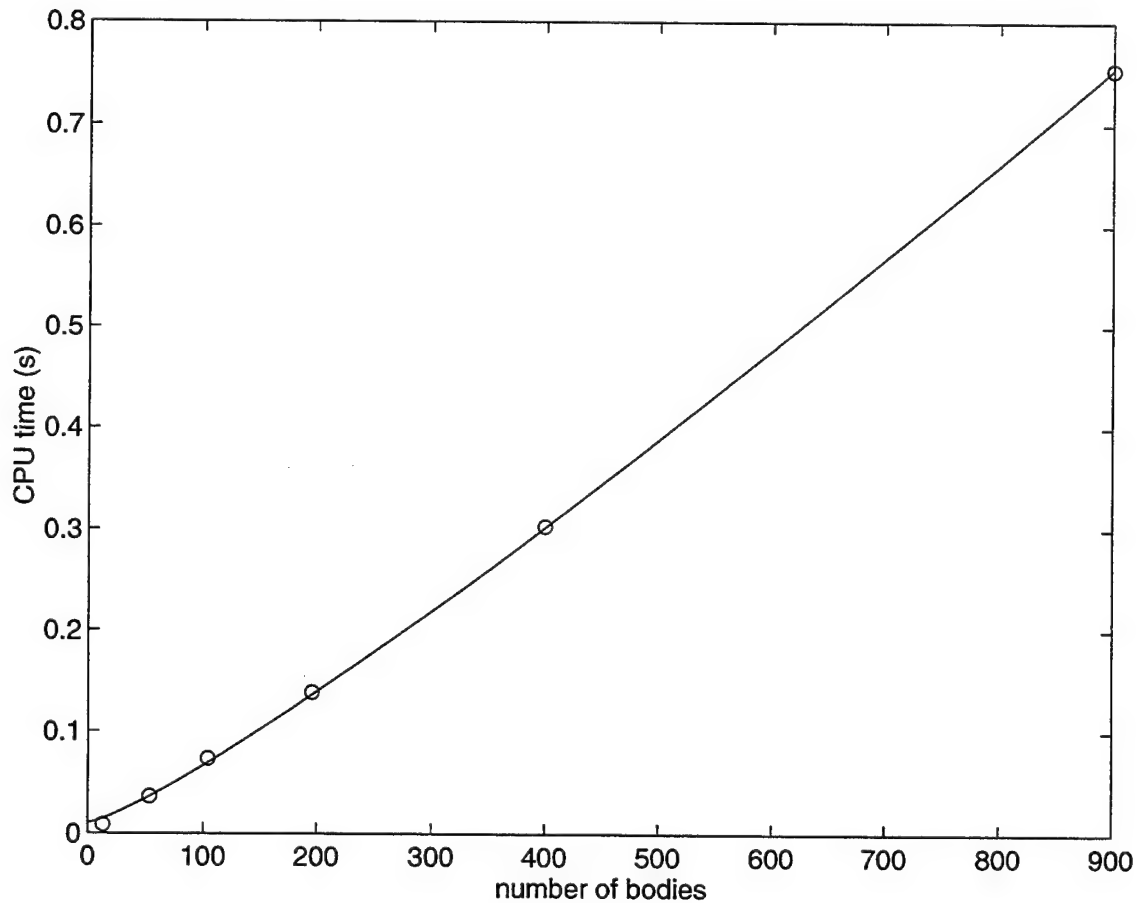


Figure 7.5: Contact detection CPU time for the algorithm with sorting procedure. Computational data ( o ); regression analysis (—).

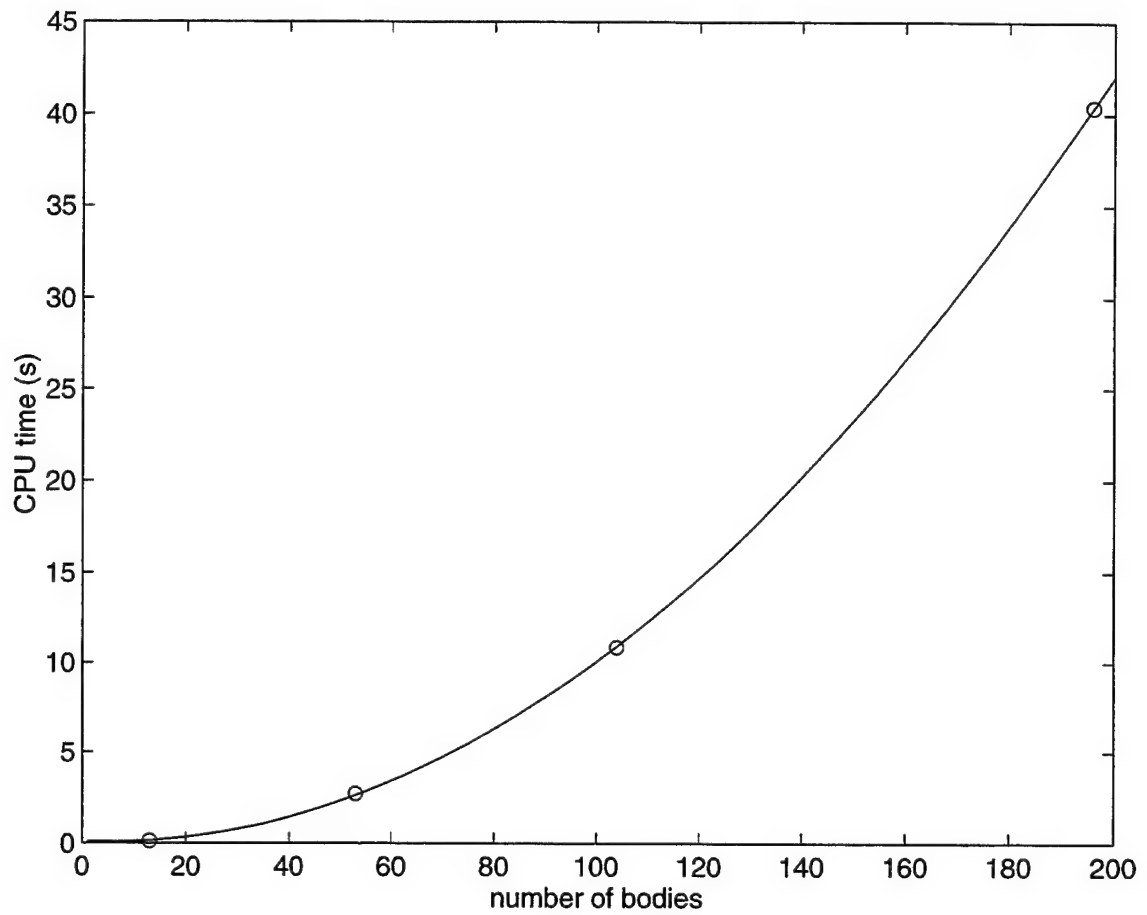


Figure 7.6: Contact detection CPU time for the algorithm with no sorting procedure. Computational data ( o ); regression analysis (—).

$N$	$\Lambda_{\text{sorting}}$	$\Lambda_{\text{no sorting}}$
13	0.0012	0.74
53	0.0418	2.91
104	0.0424	5.60
196	0.0400	10.49
400	0.0413	-
900	0.0476	-

Table 7.3: Computing times in seconds for various numbers of bodies.

procedure is also of interest, since without the addition of a sorting phase the detection algorithm would dominate the CPU time for large  $N$ . We denote this ratio by  $\Lambda$ , i.e.,

$$\Lambda_{\text{case}} = \frac{\text{CPU}_{\text{detection}}^{\text{case}}}{\text{CPU}_{\text{total}}^{\text{case}} - \text{CPU}_{\text{detection}}^{\text{case}}}, \quad (7.1)$$

where *case* refers to *sorting* or *no sorting*. In Table 7.3 we show these values for different number of bodies.

One may observe that the ratio  $\Lambda$  increases linearly with  $N$  for the non-sorting algorithm, while the value is independent of  $N$  when sorting is added to the search algorithm.

# Chapter 8

## Numerical Examples

### 8.1 Introduction

In this section we present a variety of numerical simulations with which we evaluate the performance of the various schemes presented in this work. The three sections detail a variety of benchmark problems to test the correctness of the solutions and some other problems to compare the quality of the solutions with other more conventional contact/impact schemes. The last section primarily focuses on the versatility of the sort/search scheme of the contact detection part of the algorithm in cases of multibody problems. The algorithms formulated in this work have been implemented in the general finite element code FEAP developed by Professor Robert L. Taylor of the University of California at Berkeley.

### 8.2 Frictionless contact simulations

#### 8.2.1 Impact of a rod on a rigid wall

The purpose of this simulation is to show the important role that the energy restoring property of our scheme has on the quality of the solution and on the overall stability of the simulation. The contact problem of elastic bodies may be thought of as consisting of two

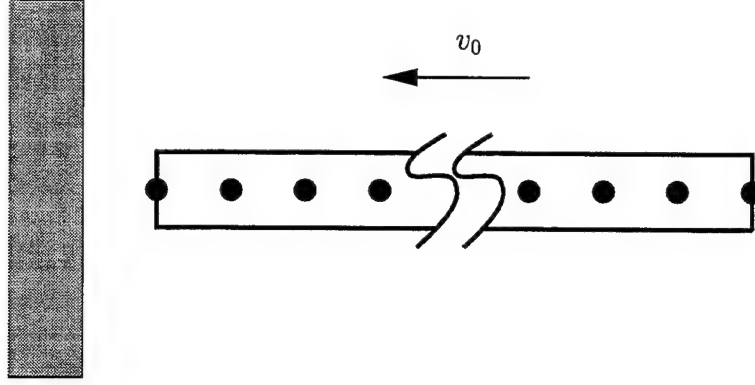


Figure 8.1: Impact of an elastic rod against a rigid wall. Schematic drawing of the problem.

parts: the continuum part, which deals with the elastic deformation, and the contact part, which deals with the unilateral constraint (impenetrability).

We solve the continuum problem using the conservative scheme presented in Chapter 3 and developed by SIMO & TARNOV [30], and compare these results with the ones obtained through HHT scheme, also known as the  $\alpha$ -method, presented in HILBER ET AL [13]. When solving the contact part of the problem we use the restoring scheme presented in Chapter 5 and the mid-point contact scheme developed by LAURSEN & SIMO [20].

The HHT scheme is dissipative for linear problems, hence unconditionally stable, but it this dissipative property for general nonlinear problems. Thus, in the case of a contact problem, one may find an increase in energy if the scheme is not used in combination with an adequate contact algorithm to deal with the nonlinearity inherent to the contact constraint.

Let us consider the one dimensional model of a rod impacting a rigid wall which can be solved using various combinations of continuum and contact algorithms.

The problem is sketched in Figure 8.1. The rod used in the simulations has length  $L = 1.0$  and is discretized with 100 isoparametric elements. The Young's modulus is  $E = 1.0$  and the density is  $\rho = 1.0$ . The initial velocity before impact is  $v_0 = -0.5$ . The rod is initially at a distance of  $7.5 \times 10^{-3}$  from the wall. The wave velocity within the bar is  $c = \sqrt{\frac{E}{\rho}}$  and the simulation is performed using a discrete  $CFL = \frac{c\Delta t}{h} = 2$ , where  $h$  is the element length in the finite element discretization. Analytical calculations show that the rod will

remain in contact with the rigid wall for a period of time  $\mathcal{T} = \frac{2L}{c}$ , so that in our problem we have  $\mathcal{T} = 2$ . The simulations should then yield  $g = 0$  and  $\dot{g} = 0$  for  $t \in [\epsilon, \epsilon + 2]$ , where  $\epsilon = \frac{7.5 \times 10^{-3}}{0.5}$  (the time the rod takes to reach the wall from its initial position). The simulations are performed with a time step  $\Delta t = 0.0025$ ; since  $\mathcal{T} \gg \Delta t$ , the imposition of the gap velocity constraint becomes necessary.

We assume a linear model so that we only introduce the nonlinearity arising from the contact phenomenon. We consider the three parameter family of time discretizations (see HILBER ET AL [13]) to accomodate dissipative schemes when we deal with the continuum problem. Thus, we have

$$\mathbf{f}_{c,n+\frac{1}{2}} = \mathbf{M}\mathbf{a}_{n+1} + \mathbf{K}[\alpha\mathbf{d}_{n+1} + (1 - \alpha)\mathbf{d}_n], \quad (8.1)$$

$$\mathbf{d}_{n+1} = \mathbf{d}_n + \Delta t \mathbf{v}_n + \frac{1}{2} \Delta t^2 [2\beta \mathbf{a}_{n+1} + (1 - 2\beta)\mathbf{a}_n], \quad (8.2)$$

$$\mathbf{v}_{n+1} = \mathbf{v}_n + \Delta t [\gamma \mathbf{a}_{n+1} + (1 - \gamma)\mathbf{a}_n], \quad (8.3)$$

where  $\mathbf{M}$  denotes the standard mass matrix and  $\mathbf{K}$  denotes the usual stiffness matrix of linear elasticity. The equilibrium equation 8.1 is written in the form presented in SIMO ET AL [31], where the parameter  $\alpha$  in [13] is equivalent to  $\alpha - 1$  in [31].

We also consider a standard mid-point contact scheme by imposing the contact constraint at  $t_{n+\alpha}$ , where  $\alpha$  is consistent with to the one used in equation 8.1. The contact pressure  $t_N$  then takes the following form:

$$t_N = \kappa_N g_{n+\alpha}, \quad (8.4)$$

where  $g_{n+\alpha}$  is the (real) gap calculated using a closest point projection in the configuration at time  $t_{n+\alpha}$ .

We consider the following combination of schemes:

**Algorithm 1** Trapezoidal rule:  $\alpha = 1.0$ ,  $\beta = 0.25$  and  $\gamma = 0.5$  for the continuum part; and the standard penalty scheme described above for the contact part.

**Algorithm 2** Mid-point rule:  $\alpha = 0.5$ ,  $\beta = 0.5$  and  $\gamma = 1.0$  for the continuum part; and the standard penalty scheme described above for the contact part.

**Algorithm 3** HHT:  $\alpha = 0.51$ ,  $\beta = 0.555025$  and  $\gamma = 0.99$  for the continuum part; and the standard penalty scheme described above for the contact part.

**Algorithm 4** Mid-point rule:  $\alpha = 0.5$ ,  $\beta = 0.5$  and  $\gamma = 1.0$  for the continuum part; and the new **energy restoring contact scheme** for the contact part.

**Algorithm 5** Mid-point rule:  $\alpha = 0.5$ ,  $\beta = 0.5$  and  $\gamma = 1.0$  for the continuum part; and the new **dissipative contact scheme** for the contact part.

Note that the conserving elastic algorithm described in Chapter 3 yields the mid-point and trapezoidal rules when applied to a linear elastic problem, so these are energy conserving schemes. On the other hand, the HHT scheme is energy dissipative for linear problems. As we illustrate with this numerical simulation, **Algorithms 1, 2 and 3** lose the energy conserving property when the contact nonlinearity is added to the problem and treated using the standard mid-point contact penalty regularization. The results also show that when either of the continuum schemes is combined with the appropriate contact scheme, the energy conserving property remains valid.

Figure 8.2.1 shows the results for the trapezoidal rule (**Algorithm 1**) compared with the proposed scheme (**Algorithm 4**). While **Algorithm 4** successfully constrains the gap and the gap velocity during the time of contact, the trapezoidal rule evidences a marked oscillatory behaviour. This type of behaviour is also reflected in the pressure plot while **Algorithm 4** yields a good approximation of the step function. The energy predicted by **Algorithm 1** grows almost monotonically during the time of contact, while the proposed scheme (**Algorithm 4**) restores the energy after the release of the rod.

As one may observe in Figure 8.2.1, the oscillatory behaviour is repeated by the standard contact scheme (**Algorithm 2**). The energy grows when the mid point rule (energy conserving) is using in conjunction with the standard penalty scheme instead of the energy restoring one.



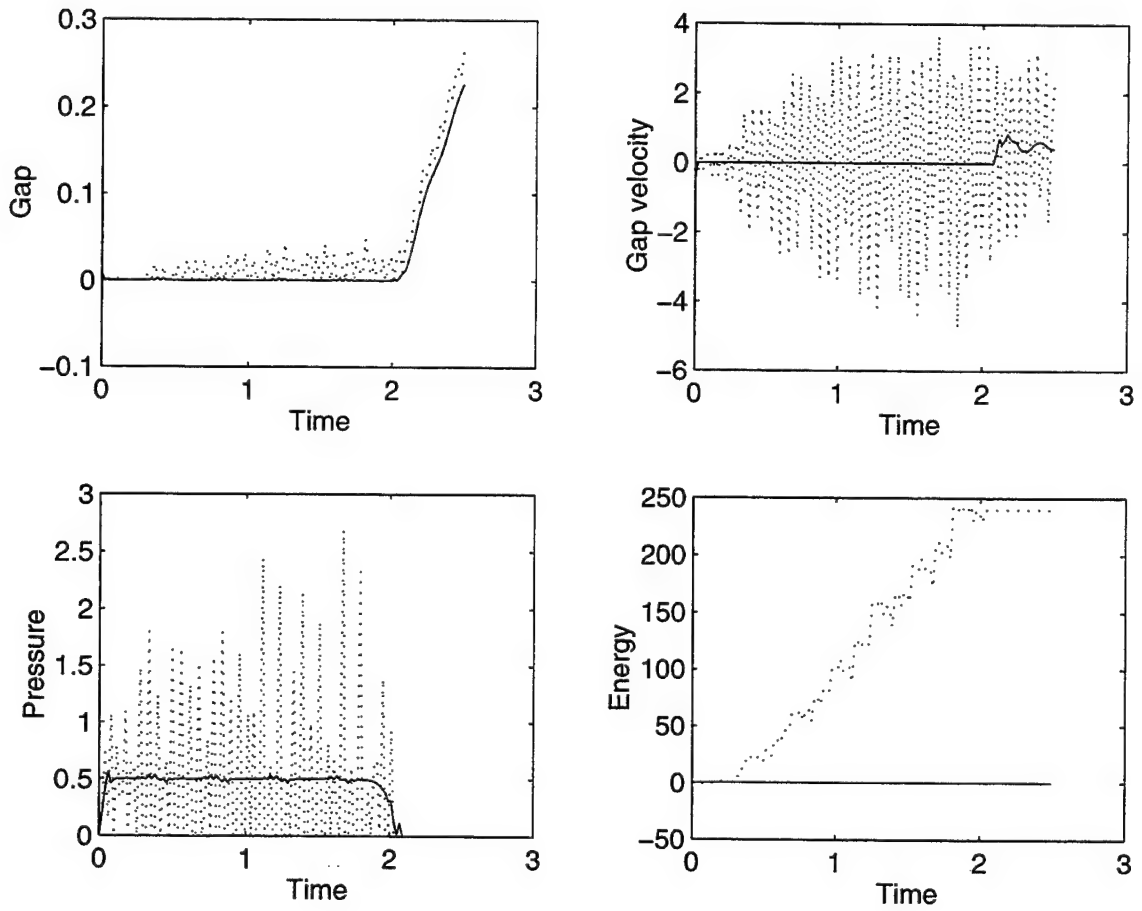


Figure 8.2: Impact of an elastic rod against a rigid wall. Plots comparing **Algorithm 1** ( $\cdots$ ) and **Algorithm 4** ( $—$ ).

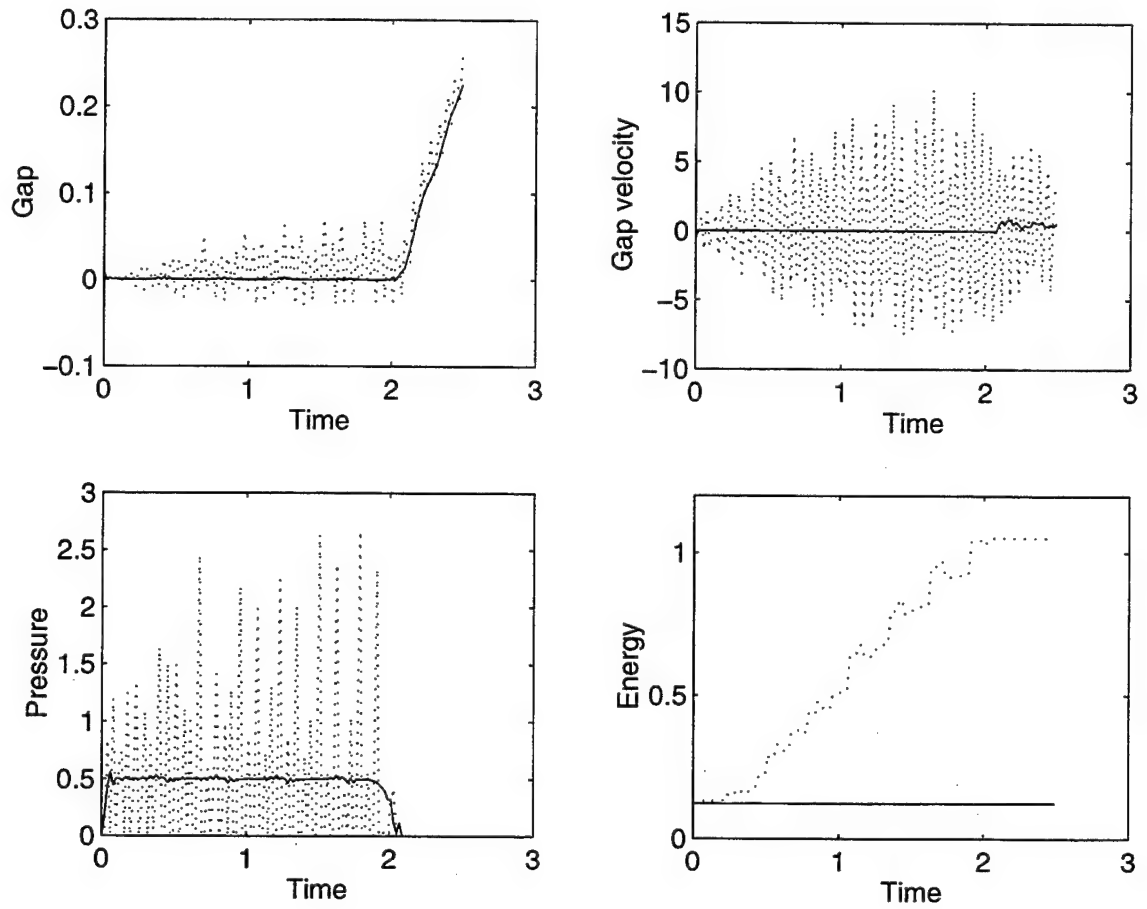


Figure 8.3: Impact of an elastic rod against a rigid wall. Plots comparing **Algorithm 2** ( $\cdots$ ) and **Algorithm 4** ( $—$ ).

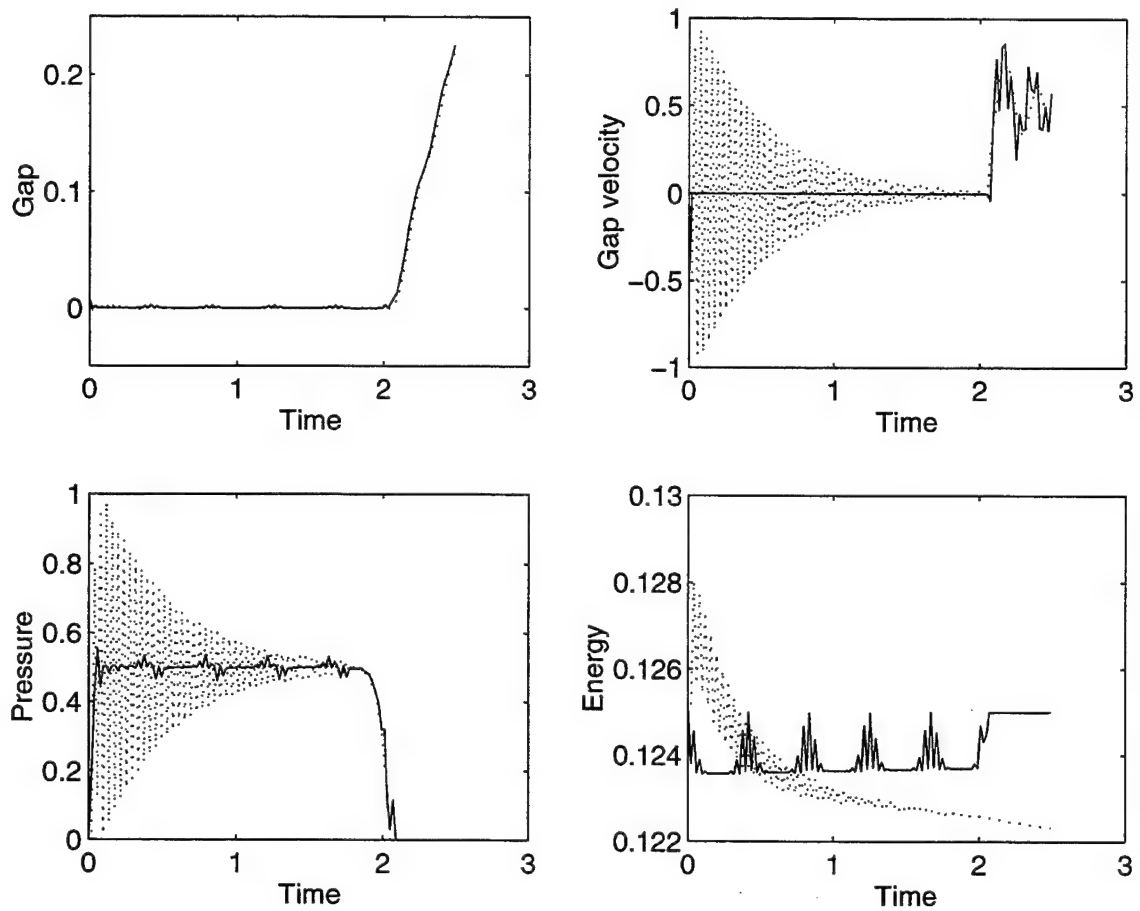


Figure 8.4: Impact of an elastic rod against a rigid wall. Plots comparing **Algorithm 3** ( $\cdots$ ) and **Algorithm 4** ( $—$ ).

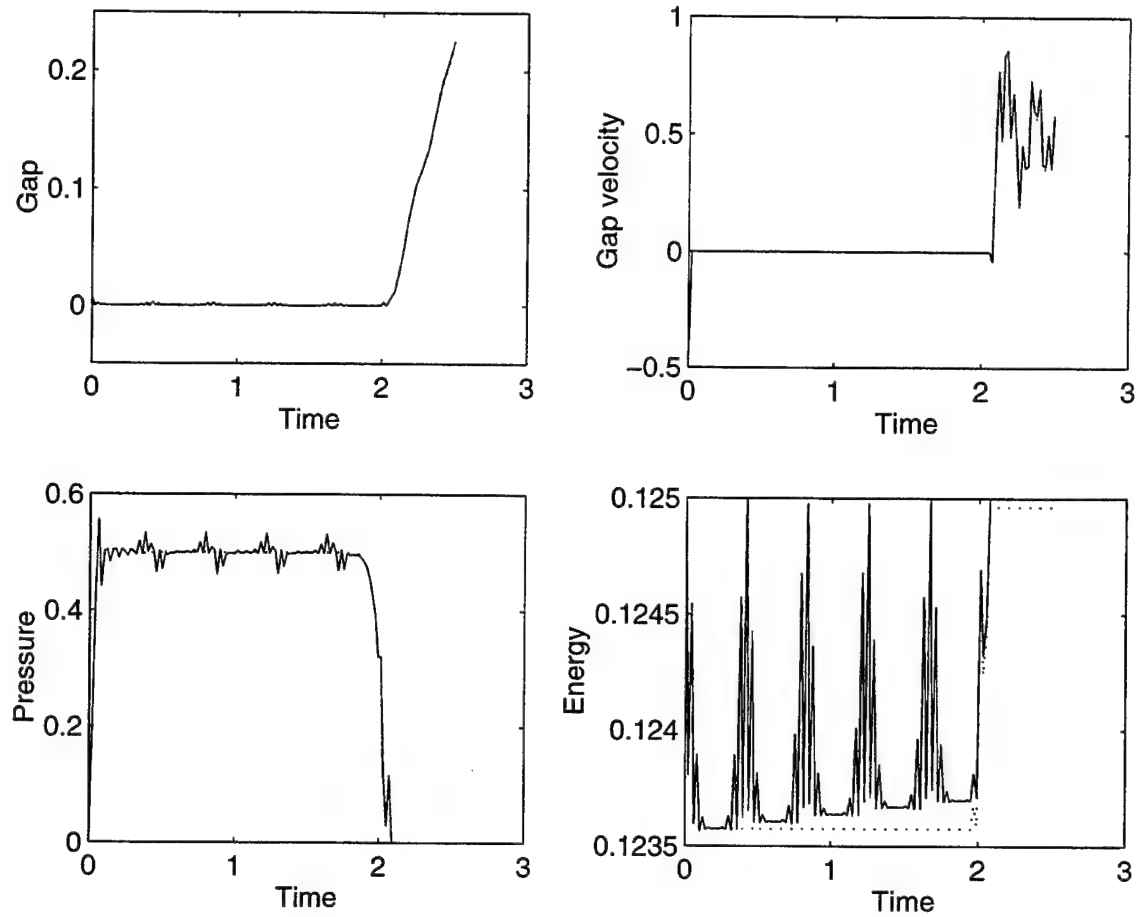


Figure 8.5: Impact of an elastic rod against a rigid wall. Plots comparing **Algorithm 5** with  $\theta = 0.9$  ( $\cdots$ ) and **Algorithm 4** ( $—$ ).

For the purpose of keeping the oscillations under control when using the standard penalty scheme, one can use a dissipative scheme to deal with the continuum part as in the case of **Algorithm 3** (see Figure 8.4). The aim of this comparison is to prove that the dissipation provided by the HHT scheme is not enough to prevent the growth of the total energy during the time of contact. As observed in Figure 8.4, the combination of schemes successfully eliminates the oscillations in the evolution of the gap function, but they are still present in the gap velocity and pressure plots. The energy increases during a short interval of time, confirming our hypothesis.

The last example, shown in Figure 8.5, demonstrates the effect of using a dissipative scheme

to deal with an impact case which has high frequency energy modes. The dissipation effectively eliminates even the small oscillations present in the energy restoring scheme. The possibility of introducing dissipation in a regulated way by changing the value of the parameter  $\theta$  makes the algorithm a very attractive choice for problems where the high frequency energy modes tend to overwhelm the results of the simulation.

### 8.2.2 Impact of two nonlinear elastic cylinders

Consider the impact of two nonlinear elastic cylinders in plane strain. The cylinders have a diameter of 2.0 and are discretized using bilinear quadrilateral finite elements. The Saint-Venant Kirchhoff material model is assumed for both cylinders. We simulate the impact of these two cylinders with two very different sets of material properties. In this way, the ability of the scheme developed in this thesis to simulate very different time scales and degrees of deformation is illustrated.

#### 8.2.2.1 Quasi-rigid cylinders

Both cylinders have Lamé constants  $\lambda = 2.0 \times 10^4$  and  $\mu = 1.0 \times 10^4$ , and density  $\rho = 1.0$ . These properties make the cylinders behave in a quasi-rigid fashion so that the duration of contact is very small compared to the overall period of simulation. Following the criteria described in Chapter 5, the satisfaction of the gap velocity constraint becomes quite unnecessary. Thus, the penalty parameters used are as follows:  $\kappa_N = 1.0 \times 10^5$  and  $m_p = 0.0$  (no satisfaction of the gap velocity constraint). A constant time step  $\Delta t = 0.1$  is used in the simulation. The left cylinder is given an initial velocity  $v_0 = (1.0, 0.2)$ ; where thereafter a series of collisions among the cylinders and the surrounding walls may be observed in Figure 8.6.

The aim of this example is to show that the unchecked growth of energy affects the dynamics of the two cylinders so that after the first impact the evolution of the system depends significantly on which algorithm is used.

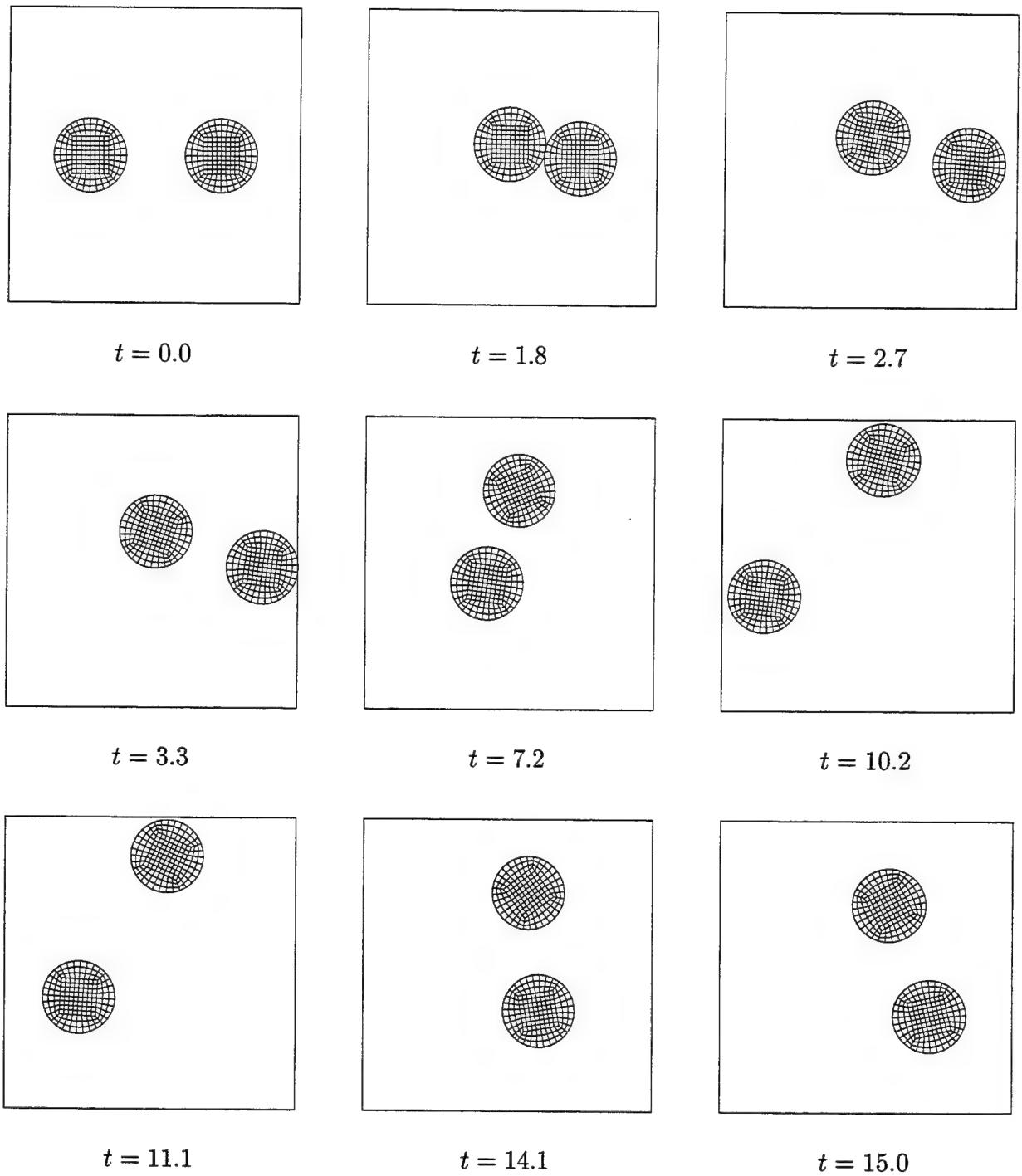


Figure 8.6: Impact of quasi-rigid cylinders. Evolution of the system

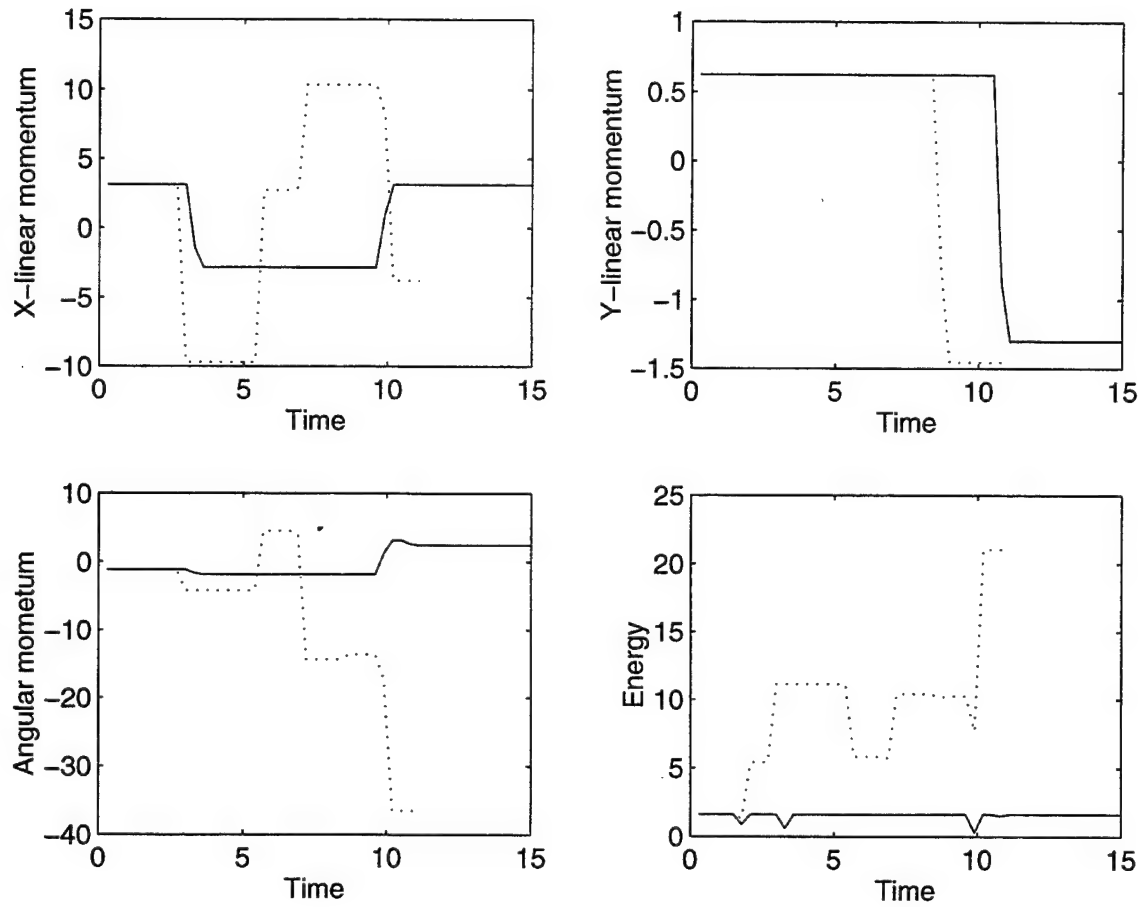


Figure 8.7: Impact of quasi-rigid cylinders. Evolution of the linear momenta, angular momentum and the energy of the system. Plots comparing **Algorithm 2** ( $\cdots$ ) and the proposed scheme ( $—$ ).

In Figure 8.7 we plot the linear momenta, angular momentum and the energy evolution for the system, using the proposed scheme and **Algorithm 2** (which uses the mid-point rule to impose the contact constraint). One may observe that the first impact occurs between the two cylinders at approximately  $t = 1.8$ , when the results show a jump in the energy evolution for the mid-point rule. The proposed scheme shows a small decrease in energy during the contact period before the energy is restored to its original value. The second impact of the problem occurs at approximately  $t = 2.8$  between one cylinder and a rigid wall. Theoretical results predict that the linear momentum in the  $x$  direction should change sign with a small variation in its magnitude while the linear momentum in the  $y$  direction should remain unchanged. In the linear momentum plots we observe that this behaviour is captured by our scheme. With the mid-point rule, the linear momentum in the  $x$  direction changes sign but its value is almost doubled. At this instant, the energy shows a marked increase.

The trajectory of the cylinders with the standard mid-point rule scheme is distinctly different from the one shown in Figure 8.6, so the instants at which impact occurs are also quite different, as evidenced by the plots in Figure 8.7.

### 8.2.2.2 Soft cylinders

The second set of material parameters for both cylinders are as follows: Lamé constants  $\lambda = 130.0$  and  $\mu = 43.33$ , and density  $\rho = 8.93$ . With this set of values, the cylinders will undergo large deformations and a long contact duration. The left cylinder is given an initial velocity  $v_0 = (1.0, 0.1)$ .

The evolution of the system is shown in Figure 8.8. One may observe that the time interval of interest in this case is of the same order as the contact time. Also, the amount of deformation of the cylinders is significant.

Figure 8.9 shows a significant increase of energy during the contact time when predicted by the mid-point scheme, whereas the proposed scheme restores the energy to its initial value at the end of the contact interval.



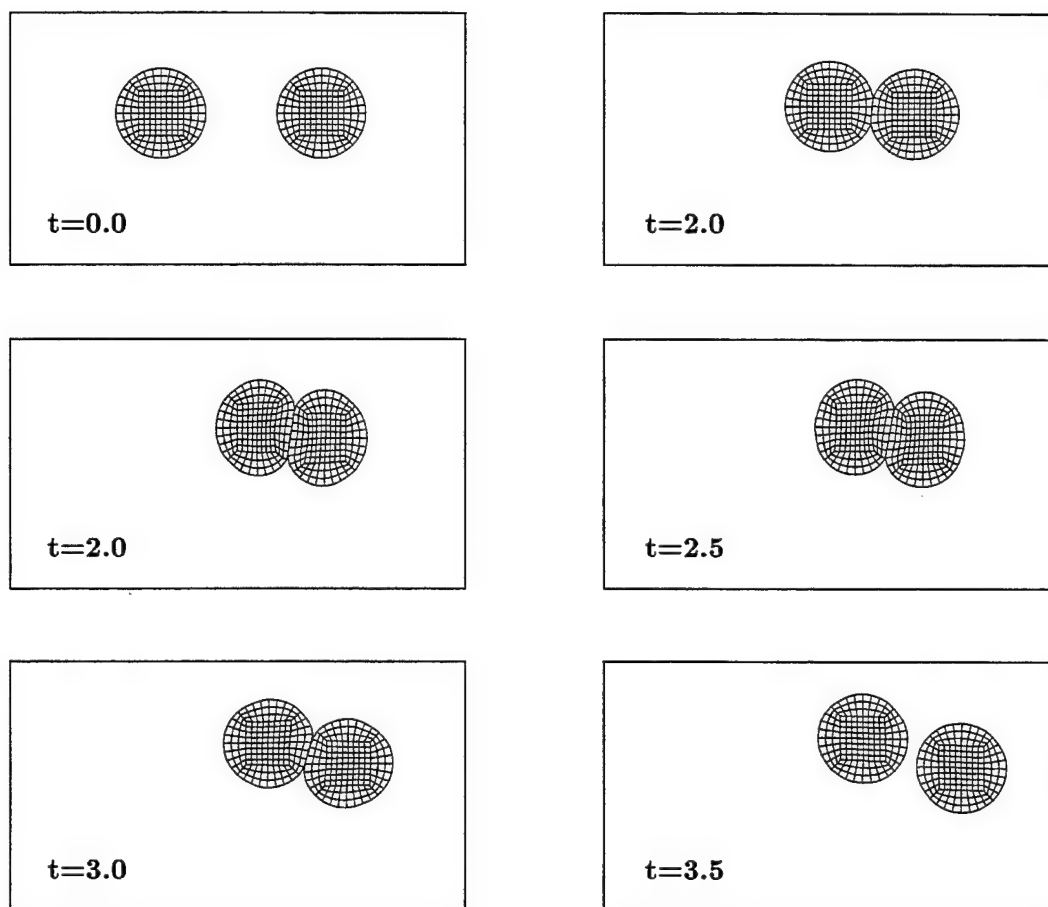


Figure 8.8: Impact between two soft elastic cylinders. Evolution of the dynamical system.

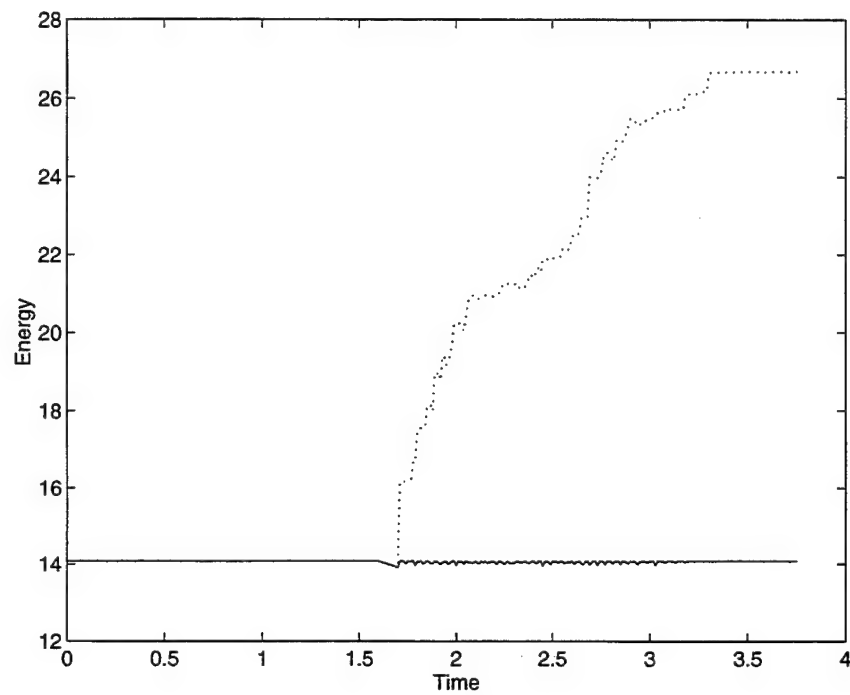


Figure 8.9: Impact between two soft elastic cylinders. Energy evolution of the dynamical system. Plots comparing **Algorithm 2**( $\cdots$ ) and **Algorithm 4**( $—$ ).

## 8.3 Frictional contact simulations

### 8.3.1 Impact of a cylinder against a rigid wall

We present in this section the results obtained by means of the proposed scheme for the problem of an elastic cylinder impacting a rigid wall at an angle. The problem has been solved using both frictionless and frictional conditions.

We have used a Saint-Venant Kirchhoff model with Lamé constants  $\lambda = 130.0$  and  $G = 43.33$ , and a density  $\rho = 8.93$ . The cylinder has a radius  $R = 1.0$  and is given an initial velocity  $v_0 = (0.4, -0.4)$ .

In the frictional case the friction constant  $\mu = 0.2$  has been used, with the following penalty parameters:  $\kappa_N = \kappa_T = 10^4$  with  $\theta = 0.5$ . The continuum contributions of the problem were solved using the energy conserving scheme detailed in [30].

To compare the overall performance of our scheme, we have also solved the problem using the standard mid-point rule for the contact and friction contributions and the energy conserving scheme for the continuum contributions.

Figures 8.10 and 8.11 show the results of the proposed scheme for three instances of the frictionless and frictional problem, respectively. Notice the absence of rotation in the frictionless case.

We observe that the mid-point rule yields an increase of energy, as shown in Figure 8.12 for the frictionless case. An increase is predicted even when there is friction involved (Figure 8.13), which should add dissipation to the system. Therefore, we conclude that the use of the dynamic contact quantities, for both the frictionless and the frictional case, contribute to the overall stability of the contact scheme.

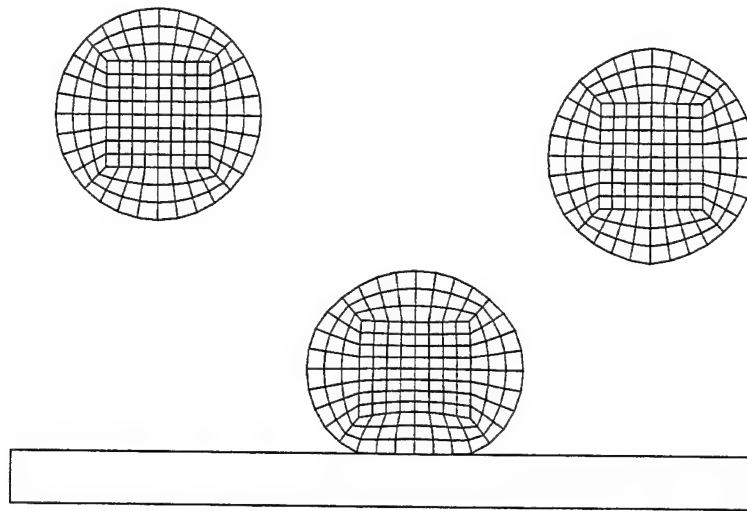


Figure 8.10: Impact of an elastic cylinder against a rigid wall. Three instances of the evolution of the frictionless case, at times  $t = 0, 6, 12$ .

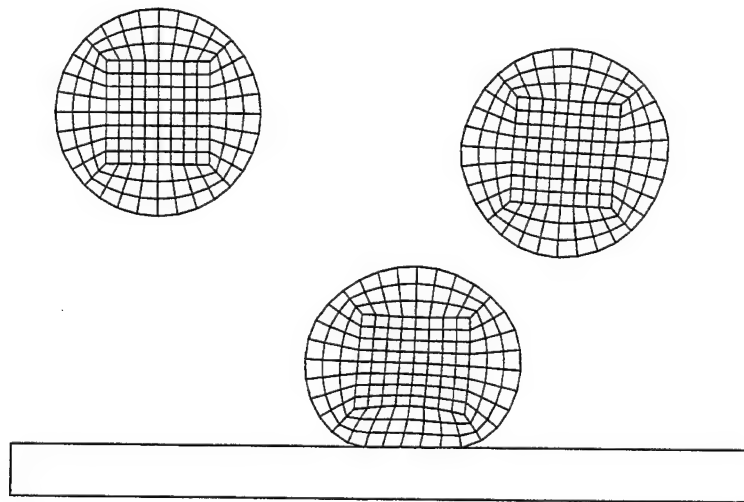


Figure 8.11: Impact of an elastic cylinder against a rigid wall. Three instances of the evolution of the frictional case ( $\mu = 0.2$ ), at times  $t = 0, 6, 12$ .

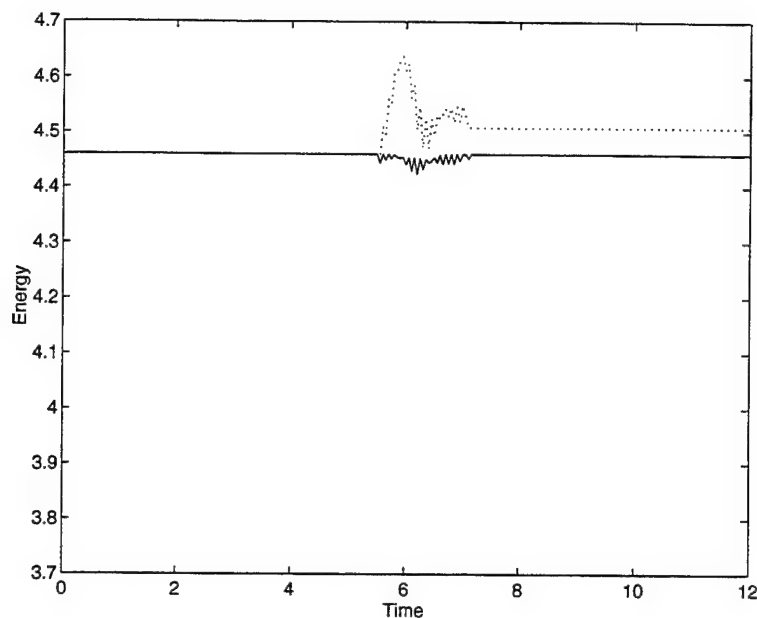


Figure 8.12: Impact of an elastic cylinder against a rigid wall. Energy evolution of the frictionless case. Plots comparing the proposed scheme (—) and the mid-point rule (···).

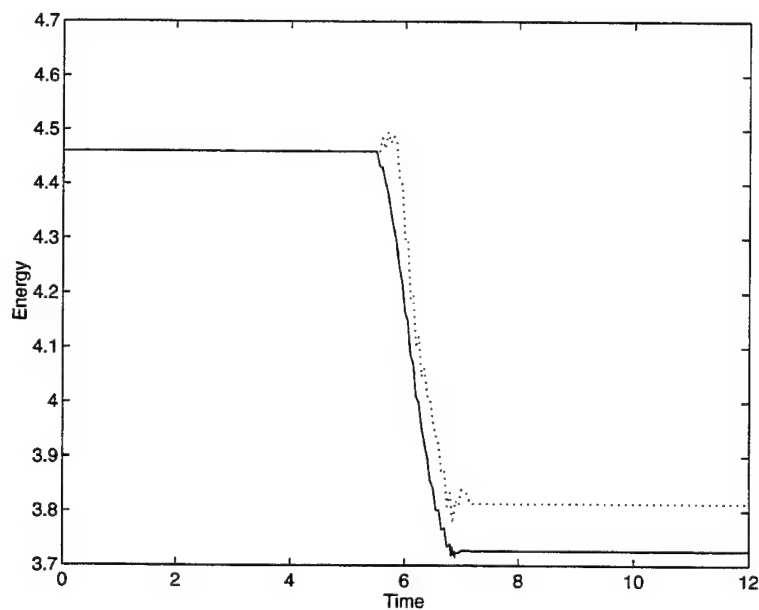


Figure 8.13: Impact of an elastic cylinder against a rigid wall. Energy evolution for the frictional case ( $\mu = 0.2$ ). Plots comparing the proposed scheme (—) and the mid-point rule (···).

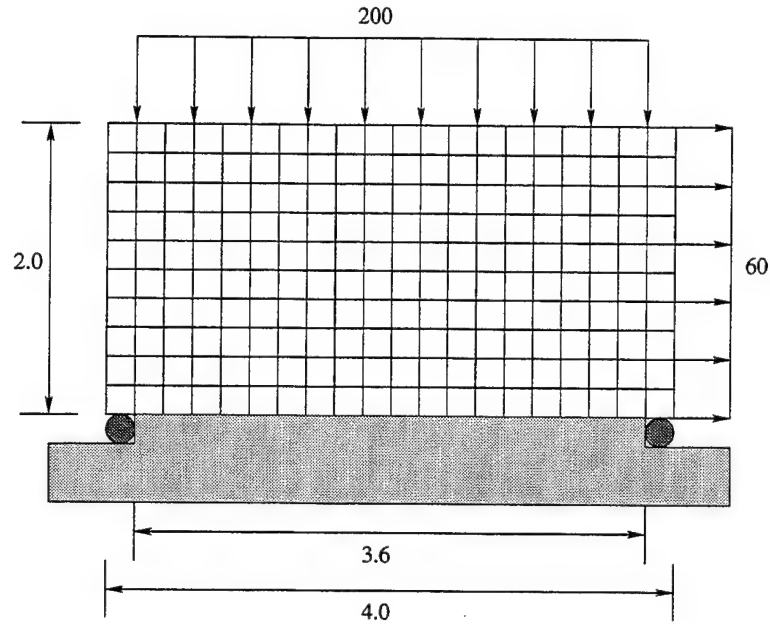


Figure 8.14: Forging of an elastic block against a rigid foundation. Schematic drawing of the problem.

### 8.3.2 Forging of an elastic block against a rigid foundation

The purpose of this simulation is to show the performance of our scheme for a frictional problem and to compare the results with other frictional schemes. In addition, this example shows that the proposed scheme can be used not only in dynamic problems but also in quasi-static problems.

Consider the problem of an elastic block pressed against a rigid foundation. The block is pulled by a tangential force uniformly distributed along one of the sides of the block, as shown in Figure 8.14.

The problem has been solved using bilinear quadrilateral elements within the linear elastic formulation (infinitesimal deformation formulation) with the following Lamé constants:  $\lambda = 576.92$  and  $G = 384.62$ .

The proposed scheme has been used with the following penalty parameters:  $\kappa_N = 10^8$  and  $\kappa_T = 10^4$  with  $\theta = 0.5$ .

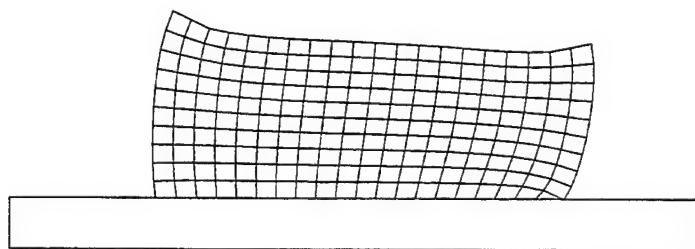


Figure 8.15: Forging of an elastic block against a rigid foundation. Deformed mesh.

The deformed block is shown in Figure 8.15. We compare the results of our scheme with the one presented in ODEN & PIRES[26], for the case with  $\mu = 0.5$  and  $\epsilon = 10^{-4}$ , where  $\epsilon = \frac{1}{\kappa_T}$ . A comparison of the stresses along the base of the block can be seen in Figure 8.16. We note that there is good agreement between the curves and the data points.

This numerical example proves not only that the scheme has good stability properties but also that the simulation results are very similar to the results of other implicit schemes which lack these properties.

### 8.3.3 Oblique impact of two infinite blocks

We present in this section the case of an oblique impact in plane strain between two linearly elastic blocks, that is, we use an elastic formulation that considers infinitesimal deformations. The top block has an initial velocity  $v_0 = (-10, -10)$  and the other has its bottom edge fixed. Figure 8.17 shows a schematic drawing of the problem. At  $t = 0.0$ , the block is resting on the bottom block (in the drawing we separated the blocks for clarity).

The penalty parameters used are as follows:  $\kappa_N = \kappa_T = 10^4$ . The problem has been simulated with the following frictional conditions:  $\mu = 0.0$  (frictionless) and  $\mu = 0.4$ . The displacement of point A (see Figure 8.17) in both directions is compared with the results obtained by CHEN & YEH[8].

Figure 8.18 displays the horizontal and vertical displacements of point A vs. time, for the

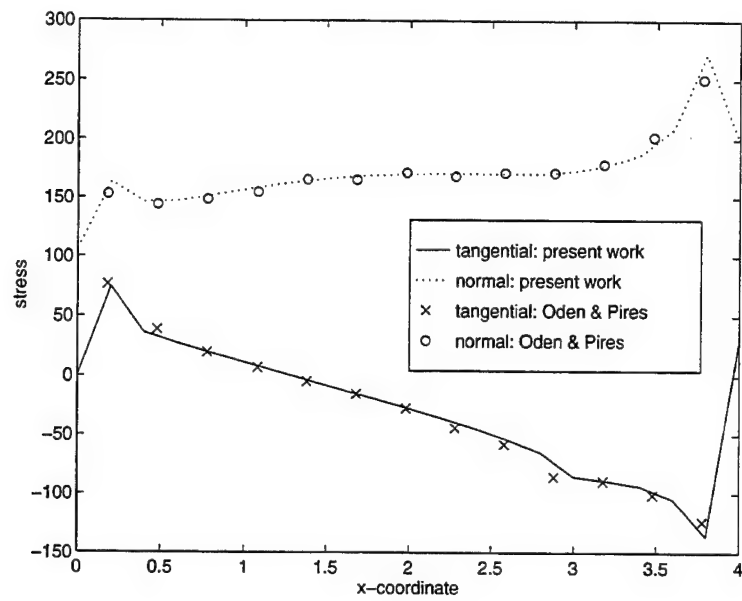


Figure 8.16: Forging of an elastic block against a rigid foundation. Stress curves along the base of the block.

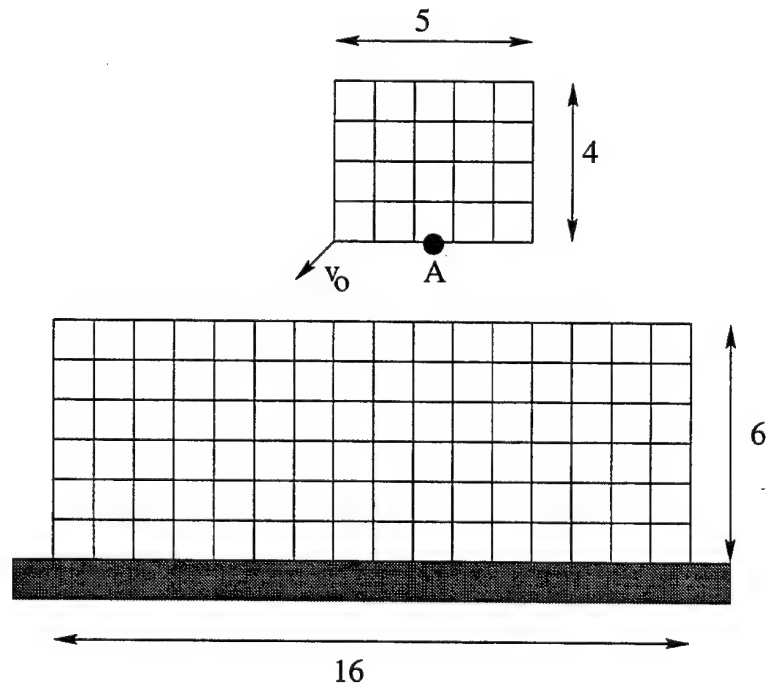


Figure 8.17: Oblique impact of two elastic blocks. Schematic drawing of the problem.



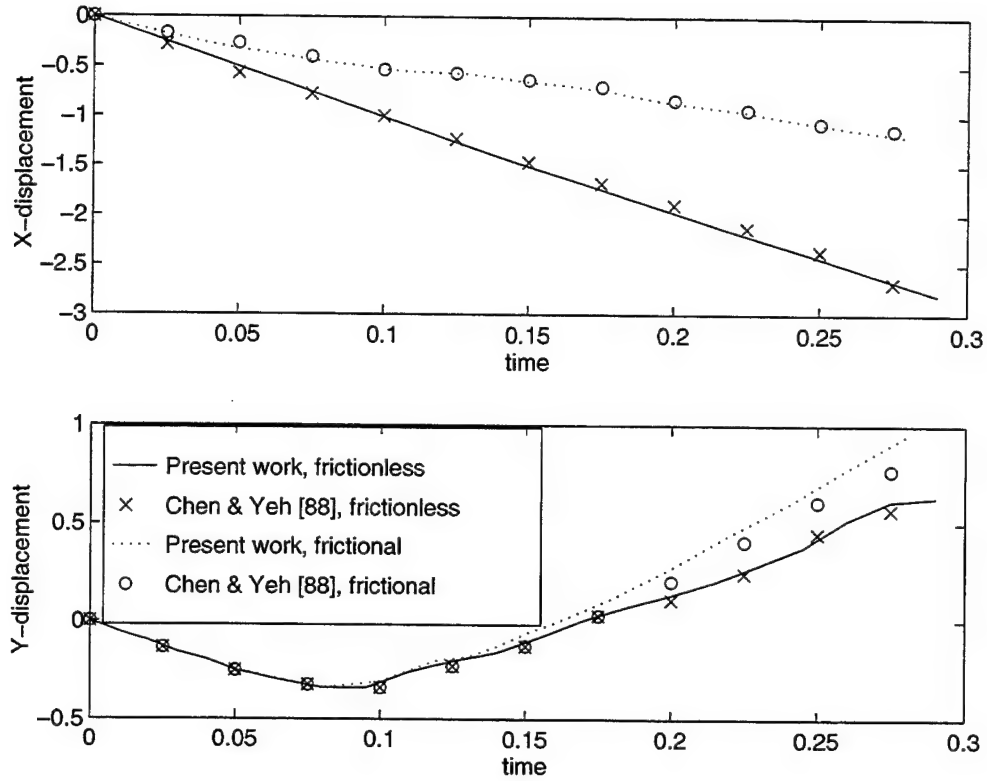


Figure 8.18: Oblique impact of two elastic blocks. Displacement of point A.

frictionless and frictional cases. As expected, the horizontal displacements are significantly reduced by the friction phenomenon, while the vertical displacements on the rebound increase when friction is present.

The results obtained by the present scheme compare well with the results from [8]. In Figures 8.19, 8.20 and 8.21 we show the deformed mesh,  $\sigma_{xx}$  and  $\sigma_{yy}$ , respectively, at time  $t = 0.1$  for the frictional case.

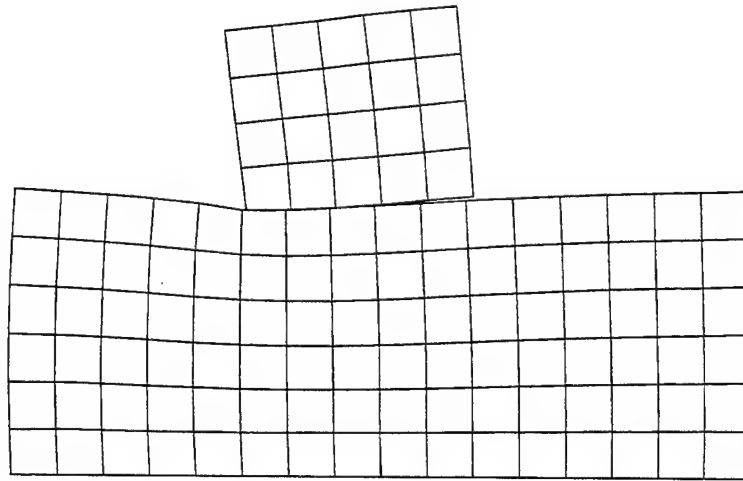


Figure 8.19: Oblique impact of two elastic blocks. Deformed mesh at  $t = 0.1$ .

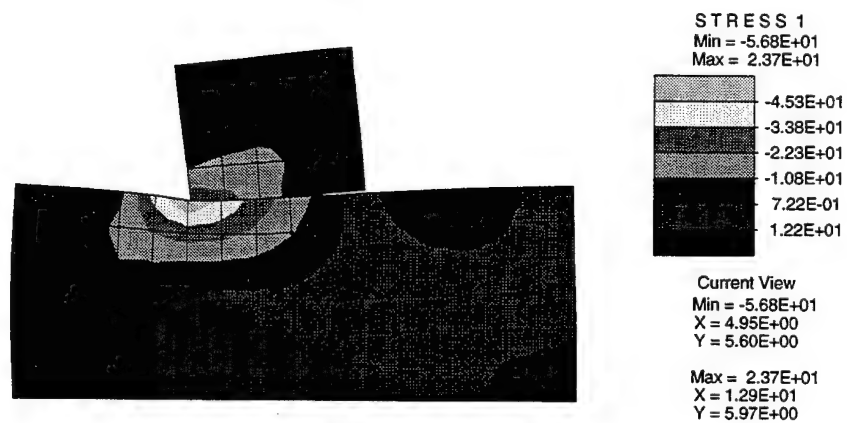


Figure 8.20: Oblique impact of two elastic blocks. Stress  $\sigma_{xx}$  at  $t = 0.1$  for the frictional case.

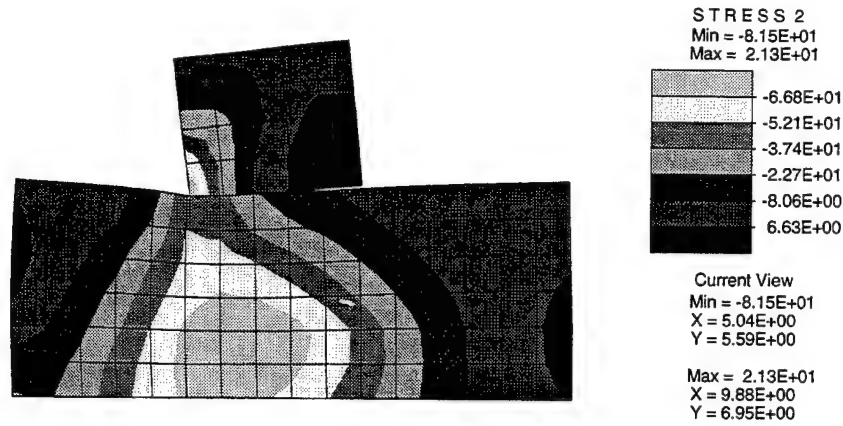


Figure 8.21: Oblique impact of two elastic blocks. Stress  $\sigma_{yy}$  at  $t = 0.1$  for the frictional case.

## 8.4 Multibody contact

### 8.4.1 Impact of 9 elastic disks

We show the numerical results of the impact of 9 disks enclosed within four walls (see Figure 8.22). The disks have Lamé constants  $\lambda = 130.0$  and  $G = 43.33$ , and density  $\rho = 8.93$ . The disk in the lower left corner is given an initial velocity  $\mathbf{v}_0 = (0.5, 0.5)$  and the disk in the upper right corner is given an initial velocity  $\mathbf{v}_0 = (-0.5, -0.5)$ . The evolution of the energy of the system is shown in Figure 8.23.

During the simulation, the energy calculated using the mid-point rule increases rapidly because of the many collisions. In contrast, the energy calculated using the proposed scheme is restored to its original value after each collision. Also, the simulation using the mid-point rule blows up.

The motion of the bodies may be observed in Figure 8.22. The choice of material properties and initial velocity of the system causes the bodies to suffer significant deformations. Therefore, this example shows the implementation using the proposed scheme to be effective in the simulation of multiple collisions among bodies undergoing finite deformations and large rotations.

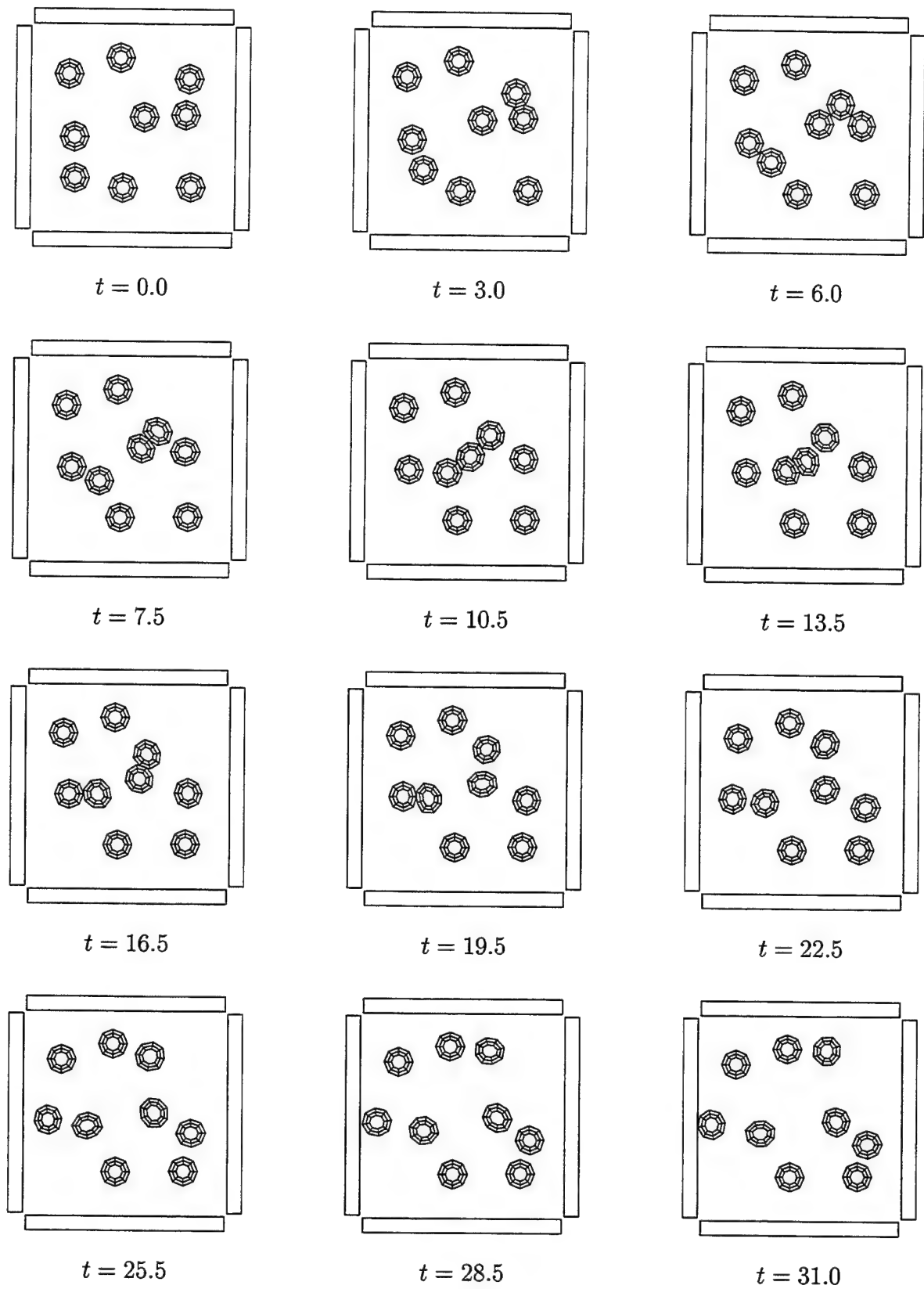


Figure 8.22: Impact of 9 elastic disks. Evolution of the system.

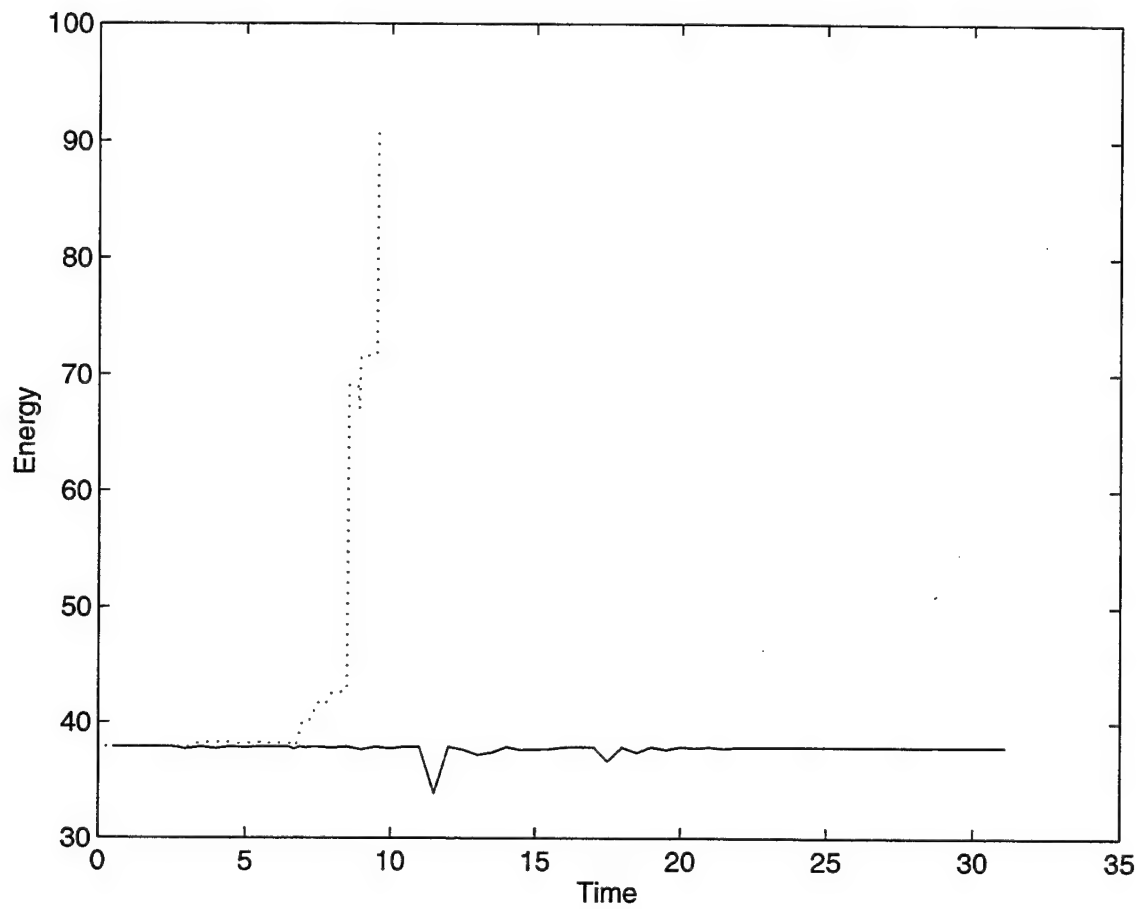


Figure 8.23: Impact of 9 elastic disks. Evolution of the energy of the system. Plots comparing **Algorithm 2**( $\cdots$ ) and the proposed scheme( $—$ ).

### 8.4.2 Impact of 49 quasi-rigid disks

In this numerical example we show simulation results of the impact of 49 elastic disks enclosed within four rigid walls. The disks have Lamé constants  $\lambda = 2000.0$  and  $G = 1000.0$ , and density  $\rho = 1.0$ . The left-most column of disks is given an initial velocity  $v_0 = (0.5, 0.1)$ . Figures 8.4.2 and 8.4.2 show the evolution of the system. Notice that clusters of bodies are formed at various instants, i.e. a particular body may be in contact with many others at certain time steps.

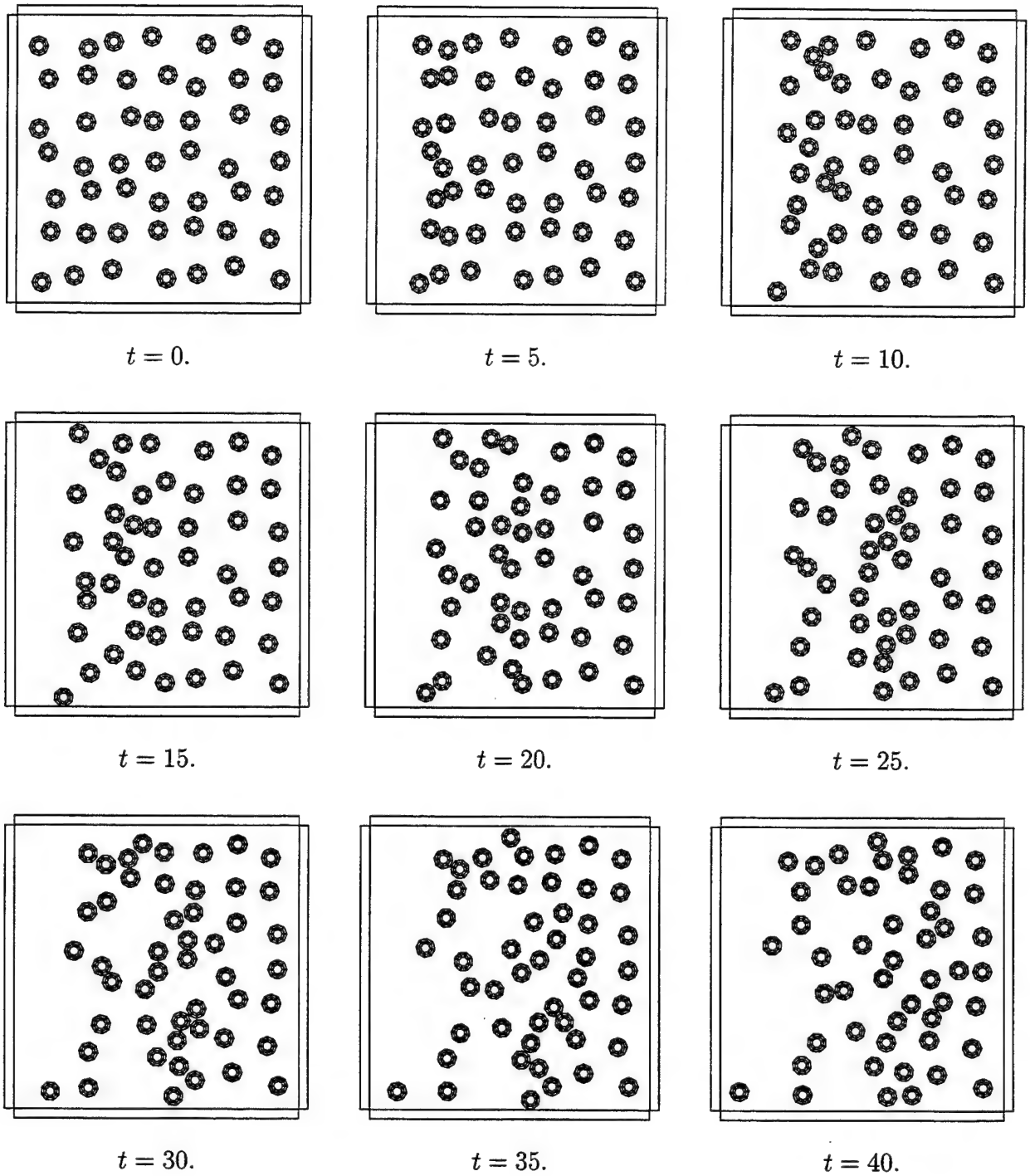


Figure 8.24: Impact of 49 quasi-rigid disks. Evolution of the system,

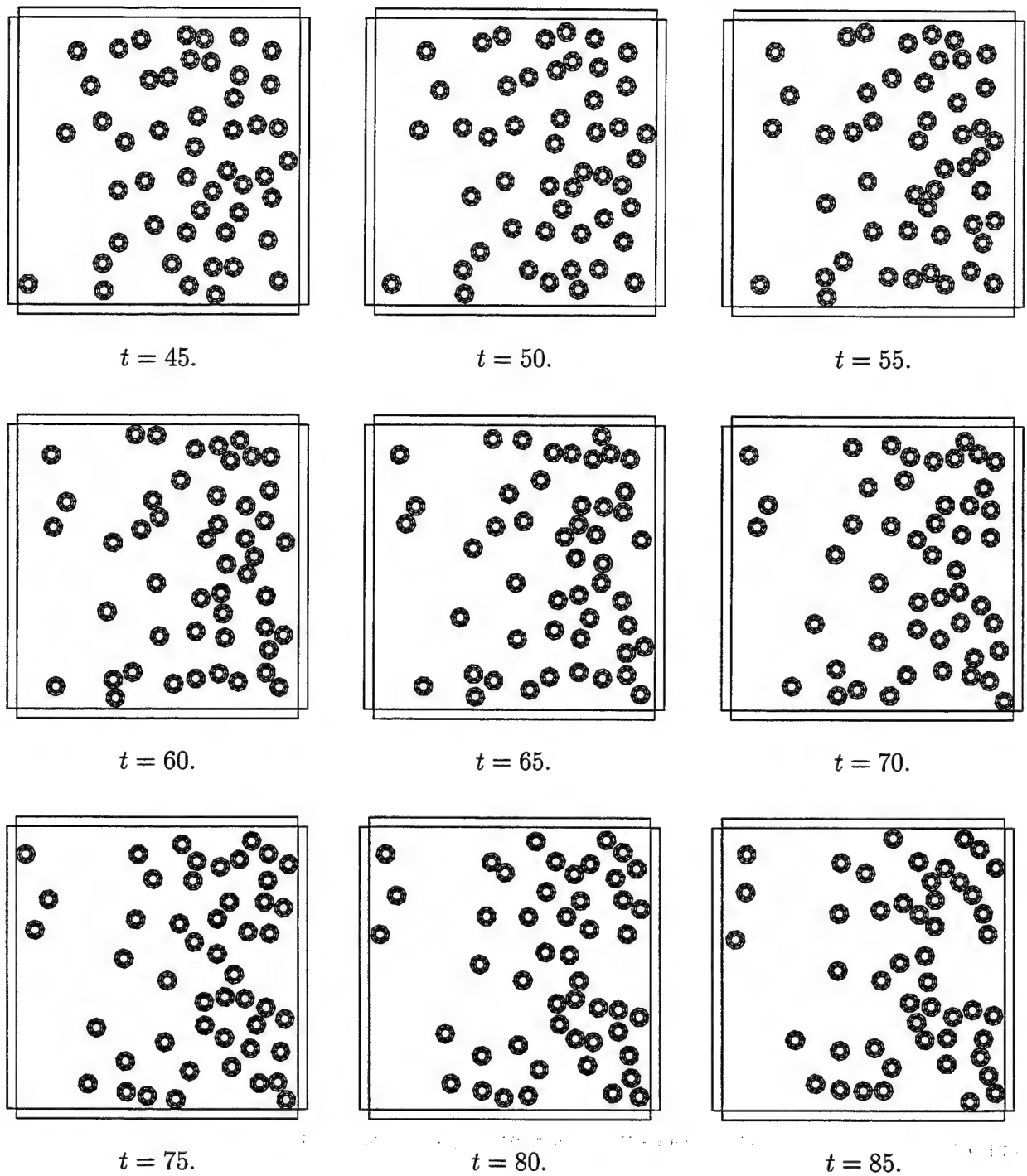


Figure 8.25: Impact of 49 quasi-rigid disks. Evolution of the system (continued).



# Chapter 9

## Conclusions

### 9.1 Closure

In this dissertation, the formulation of a new implicit stable time-stepping algorithm for dynamic contact problems has been presented. The salient points of the proposed scheme are the following:

1. In the frictionless case, the new contact scheme unconditionally inherits the conservation properties of the underlying continuum problem, thus simulating the correct physical system. We have also shown, by means of numerical examples, that the results obtained may be significantly different if non-conserving schemes are used, even if no numerical blow-up occurs. With almost no modifications, our scheme can also be used also in quasi-static simulations.
2. In the frictional case, the scheme is proven to be unconditionally dissipative thus imitating the real physical system. It has been shown, by means of numerical examples, that standard frictional schemes may predict energy increases, thus yielding unphysical behaviour. Furthermore, our frictional scheme can also be used in quasi-static problems.

3. Using penalty methods, we are able to enforce the constraint on the gap velocity, in addition to satisfying efficiently the usual impenetrability constraint. This enhancement does not affect the previously mentioned conserving properties of the proposed scheme. The enforcement of this constraint enables the proposed scheme to circumvent some of the difficulties present in short-duration simulations in which oscillatory behaviour tends to overwhelm the real solution.
4. Since the energy is under control at all times, we have an (energy) stable algorithm. This property yields a robust scheme which is easily implementable, with only slight changes, into the standard penalty formulations.
5. We have also discussed the introduction of positive high frequency energy dissipation for short-duration simulations. The amount of dissipation can be regulated depending on the problem at hand, so as to eliminate the desired degree of oscillation in the results. This enhancement requires only a slight modification of the original energy restoring scheme.
6. The implementation of the proposed scheme in a multibody formulation has been described in detail. Several concepts from the discrete element method were found to be versatile and adaptable to a finite element contact formulation, in particular the contact detection algorithms. The introduction of sorting procedures prior to the contact resolution phase, namely the closest point procedure, proved to be efficient when dealing with multiple bodies.
7. Several numerical examples have been presented to show improved results when using the new scheme, demonstrating the value of satisfying the conservation laws of the dynamical system in question. The computational effort to simulate these problems was not higher when using our scheme than when using more conventional schemes.

## 9.2 Future work

Contact problems are in fact coupled problems, in the sense that there are multiple bodies which can be simulated independently unless they come in contact with one another. In a dynamical system, keeping track of the bodies which are in contact can become cumbersome. Therefore, the use of staggered algorithms may work towards improving the efficiency of the existing algorithms. Of course, it is desirable that these staggered schemes have the same stability properties as the coupled scheme.

From a practical standpoint, adding other physical phenomena to the contact problem is an attractive option. Some examples are the simulation of heat during contact, where the source of heat may be friction or plastic deformation.

Future work may also concentrate on the development of more efficient contact detection algorithms which make use of object-oriented programming, especially in the area of multibody simulations.

# Appendix A

## Consistent Linearization of the Proposed Schemes

For two dimensional problems, we develop in this appendix the consistent linearization of the time stepping schemes we have presented in previous chapters.

In Section A.1, the linearized equations of the problem are developed and, in Section A.2 and A.3 the linearized contributions of the normal and frictional contact forces are presented respectively.

### A.1 The linearized equations

Consider the system of ordinary differential equations 4.42 and 4.43 and introduce in them the definition of the nodal momenta  $p$  defined in equation 4.44. In this way the linearization of equation 4.43 remains unchanged even with the introduction of the modified nodal linear momenta by means of the mass penalty  $m_p$  for the nodes in contact. Equation 4.43 becomes nonlinear due to this penalization thus we need to consider its linearization too. Therefore, we consider the following residuals:

$$R_d := \frac{1}{\Delta t} M (d_{n+1} - d_n) - p_{n+\frac{1}{2}}, \quad (A.1)$$

$$R_a := f_{ext}^{(n+\frac{1}{2})} + f_c^{(n+\frac{1}{2})} - f_{int}^{(n+\frac{1}{2})} - \frac{1}{\Delta t} (p_{n+1} - p_n). \quad (A.2)$$

Using the Newton-Raphson methodology, we perform the consistent linearization of the residuals A.1 and A.2, yielding

$$R_d^{(i)} + \frac{1}{\Delta t} M \Delta d_{n+1}^{(i+1)} - \frac{1}{2} \Delta p_{n+1}^{(i+1)} = 0, \quad (A.3)$$

$$R_a^{(i)} + \Delta f_c^{(n+\frac{1}{2})} - \Delta f_{int}^{(n+\frac{1}{2})} - \frac{1}{\Delta t} \Delta p_{n+1}^{(i+1)} = 0, \quad (A.4)$$

where

$$d_{n+1}^{(i+1)} = d_{n+1}^{(i)} + \Delta d_{n+1}^{(i+1)} \quad (A.5)$$

and

$$p_{n+1}^{(i+1)} = p_{n+1}^{(i)} + \Delta p_{n+1}^{(i+1)} \quad (A.6)$$

are the updates between iterations  $(i)$  and  $(i+1)$  in time step  $[t_n, t_{n+1}]$ .

Consider the following notation

$$\Delta f_{int}^{(n+\frac{1}{2})} := K_{int}^{(i)} \Delta d_{n+\frac{1}{2}}^{(i+1)}, \quad (A.7)$$

with  $\Delta d_{n+\frac{1}{2}}^{(i+1)} = \frac{1}{2} \Delta d_{n+1}^{(i+1)}$  for the contributions of the internal elastic forces to the tangent stiffness, and

$$\Delta f_c^{(n+\frac{1}{2})} := -K_c^{(i)} \Delta d_{n+\frac{1}{2}}^{(i+1)}, \quad (A.8)$$

for the contribution of the contact forces. .

Using equation A.1, we eliminate  $\Delta \mathbf{p}_{n+1}^{(i+1)}$  from equation A.2. The introduction of equations A.7 and A.8, leads to

$$\left[ \frac{1}{2} \left( \mathbf{K}_{int}^{(i)} + \mathbf{K}_c^{(i)} \right) + \frac{2}{t^2} \mathbf{M} \right] \Delta \mathbf{d}_{n+1}^{(i+1)} = \mathbf{R}_a^{(i)} - \frac{2}{t} \mathbf{R}_d^{(i)}, \quad (\text{A.9})$$

The previous equation yields  $\Delta \mathbf{d}_{n+1}^{(i+1)}$ , which in turn is used to calculate  $\Delta \mathbf{p}_{n+1}^{(i+1)}$ . We are able to find the expression for the update of the nodal velocities  $\mathbf{v}_{n+1}^{(i+1)}$  using the definition of the modified nodal momenta given in equation 5.38 and is given by

$$\mathbf{p}_{n+1}^{(i+1)} = \underbrace{\left( \mathbf{M} + \sum_{s=1}^{n_{slave}} m_{s,n+1}^{(i+1)} \widehat{\mathbf{G}}_{s,n+1}^{(i+1)} \otimes \widehat{\mathbf{G}}_{s,n+1}^{(i+1)} \right)}_{=: \mathbf{M}^{*(i+1)}} \mathbf{v}_{n+1}^{(i+1)}. \quad (\text{A.10})$$

Notice that the update equation A.10 is only applied to the nodes participating in the mass penalty regularization. For the rest of the nodes, equation A.1 is linear so that its linearization leads to

$$\begin{aligned} \mathbf{v}_{n+1}^{(i+1)} &= \mathbf{v}_{n+1}^{(i)} + \Delta \mathbf{v}_{n+1}^{(i+1)} \\ &= \mathbf{v}_{n+1}^{(i)} + \frac{2}{\Delta t} \Delta \mathbf{d}_{n+1}^{(i+1)}. \end{aligned} \quad (\text{A.11})$$

**Remark A.1** An implementation avoiding the use of nodal momenta  $\mathbf{p}$  for the nodes in contact can be easily devised by considering the linearized version of equation A.10. Details are omitted.

## A.2 The contact stiffness

The linearization of the contact force  $\mathbf{f}_c$ , defined by equation 4.36 as

$$\mathbf{f}_c^{(n+\frac{1}{2})} = \sum_{s=1}^{n_{slave}} \widehat{\mathbf{f}}_{s,c}^{(n+\frac{1}{2})}, \quad (\text{A.12})$$

with  $\hat{f}_{s,c}^{(n+\frac{1}{2})} = t_N \hat{G}_{s,n+\frac{1}{2}} + t_T \hat{H}_{s,n+\frac{1}{2}}$ , is given by:

$$\Delta \hat{f}_{s,c}^{(n+\frac{1}{2})} = \sum_{s=1}^{n_{slave}} \Delta \hat{f}_{s,c}^{(n+\frac{1}{2})}, \quad (\text{A.13})$$

with

$$\Delta \hat{f}_{s,c}^{(n+\frac{1}{2})} = \underbrace{\Delta t_N \hat{G}_{s,n+\frac{1}{2}}}_{\text{material(normal part)}} + \underbrace{t_N \Delta \hat{G}_{s,n+\frac{1}{2}}}_{\text{geometric(normal part)}} + \underbrace{\Delta t_T \hat{H}_{s,n+\frac{1}{2}}}_{\text{material(tangential part)}} + \underbrace{t_T \Delta \hat{H}_{s,n+\frac{1}{2}}}_{\text{geometric(tangential part)}}, \quad (\text{A.14})$$

and we have defined the expression  $\hat{G}_{s,t}$  in equation 4.38 and the expression  $\hat{H}_{s,t}$  in 6.4. We use the following notation to rewrite equation A.14:

$$\Delta \hat{f}_{s,c}^{(n+\frac{1}{2})} = K_c \Delta d_{n+1} \quad (\text{A.15})$$

$$= K_N^{mat} + K_N^{geo} + K_T^{mat} + K_T^{geo}. \quad (\text{A.16})$$

Next we develop the linearizations for the normal and tangential contributions of the contact force to the overall consistent stiffness tangent matrix, that is  $K_c$ .

### A.2.1 The normal contact stiffness

The material part of the normal contact stiffness is obtained through the linearization of the normal contact pressure  $t_N$  and is given by

$$\Delta t_N = \begin{cases} -\frac{U'(g_{s,n+1}^d) - t_N}{g_{s,n+1}^d - g_{s,n}^d} \Delta g_{s,n+1}^d & \text{if } g_{s,n+1}^d \neq g_{s,n}^d, \\ -U''(g_{s,n+1}^d) \Delta g_{s,n+1}^d & \text{if } g_{s,n+1}^d = g_{s,n}^d, \end{cases} \quad (\text{A.17})$$

where  $\Delta g_{s,n+1}^d$  is the linearization of the dynamic gap function found in LAURSEN[18],

$$\Delta g_{s,n+1}^d = \left[ 2\widehat{G}_{s,n+\frac{1}{2}} - c_1\widehat{D}_{s,n+\frac{1}{2}} - c_2\widehat{T}_{s,n+\frac{1}{2}} \right]^T \Delta \widehat{d}_{s,n+\frac{1}{2}}, \quad (\text{A.18})$$

$$c_1 = \frac{1}{m_{11} + g_{s,n+\frac{1}{2}} \nu_{s,n+\frac{1}{2}} \cdot \tau_{n+\frac{1}{2},1}(\bar{\xi})} \left[ -\widehat{T}_{s,n+\frac{1}{2}}^T (\widehat{d}_{s,n+1} - \widehat{d}_{s,n}) + g_{s,n+\frac{1}{2}} \nu_{n+\frac{1}{2}}^T (\tau_{n+1} - \tau_n) \right], \quad (\text{A.19})$$

$$c_2 = \frac{1}{m_{11} + g_{s,n+\frac{1}{2}} \nu_{s,n+\frac{1}{2}} \cdot \tau_{n+\frac{1}{2},1}(\bar{\xi})} \left[ \nu_{n+\frac{1}{2}}^T (\tau_{n+1} - \tau_n) + \widehat{T}_{s,n+\frac{1}{2}}^T (\widehat{d}_{s,n+1} - \widehat{d}_{s,n}) \right] - \frac{\widehat{T}_{s,n+\frac{1}{2}}^T (\widehat{d}_{s,n+1} - \widehat{d}_{s,n})}{g_{s,n+\frac{1}{2}} m_{11}}, \quad (\text{A.20})$$

with

$$\widehat{D}_{s,n+\frac{1}{2}} = \begin{bmatrix} 0 \\ -N_{,\xi}^{M1}(\bar{\xi}) \nu_{n+\frac{1}{2}} \\ -N_{,\xi}^{M2}(\bar{\xi}) \nu_{n+\frac{1}{2}} \\ \vdots \end{bmatrix}, \quad \text{and} \quad \widehat{T}_{s,n+\frac{1}{2}} = \begin{bmatrix} \tau_{n+\frac{1}{2}} \\ -N^{M1}(\bar{\xi}) \tau_{n+\frac{1}{2}} \\ -N^{M2}(\bar{\xi}) \tau_{n+\frac{1}{2}} \\ \vdots \end{bmatrix}. \quad (\text{A.21})$$

Here, we have employed the notation

$$\Delta \widehat{d}_{s,n+\frac{1}{2}} = \begin{bmatrix} \Delta d_{n+\frac{1}{2}}^s \\ \Delta d_{n+\frac{1}{2}}^{M1} \\ \Delta d_{n+\frac{1}{2}}^{M2} \\ \vdots \end{bmatrix}, \quad \text{and} \quad \widehat{v}_{s,n+\frac{1}{2}} = \begin{bmatrix} v_{n+\frac{1}{2}}^s \\ v_{n+\frac{1}{2}}^{M1} \\ v_{n+\frac{1}{2}}^{M2} \\ \vdots \end{bmatrix}. \quad (\text{A.22})$$

The expression  $g_{s,n+\frac{1}{2}}$  refers to the real gap calculated through the closest point projection at the configuration at  $t_{n+\frac{1}{2}}$ ,  $\tau_{n+\frac{1}{2}}$  is the tangent vector to the master surface at the point of contact ( i.e.  $\nu_{n+\frac{1}{2}} \cdot \tau_{n+\frac{1}{2}} = 0$ ) and  $l_s$  is the length of the surface element of the master surface corresponding to the given slave node  $S$ .

Let us define the following auxiliary vectors:



$$\widehat{\mathbf{B}}_{s,n+\frac{1}{2}} := \frac{1}{m_{11} + g_{s,n+\frac{1}{2}} \nu_{s,n+\frac{1}{2}} \cdot \tau_{n+\frac{1}{2},1}(\bar{\xi})} \left[ \widehat{\mathbf{T}}_{s,n+\frac{1}{2}} + g_{s,n+\frac{1}{2}} \mathbf{D}_{s,n+\frac{1}{2}} \right], \quad (\text{A.23})$$

$$\widehat{\mathbf{N}}_{s,n+\frac{1}{2}} := \widehat{\mathbf{D}}_{s,n+\frac{1}{2}} - \tau_{n+\frac{1}{2},1} \cdot \widehat{\mathbf{B}}_{s,n+\frac{1}{2}}. \quad (\text{A.24})$$

The geometrical part of the tangent, arising from the change of normal and contact point in  $\widehat{\mathbf{G}}_{s,n+\frac{1}{2}}$ , is as follows

$$\begin{aligned} \Delta \widehat{\mathbf{G}}_{s,n+\frac{1}{2}} = & \left[ \frac{g}{l_s^2} \widehat{\mathbf{N}}_{s,n+\frac{1}{2}} \otimes \widehat{\mathbf{N}}_{s,n+\frac{1}{2}} - \widehat{\mathbf{B}}_{s,n+\frac{1}{2}} \otimes \widehat{\mathbf{D}}_{s,n+\frac{1}{2}} \right. \\ & \left. - \widehat{\mathbf{D}}_{s,n+\frac{1}{2}} \otimes \widehat{\mathbf{B}}_{s,n+\frac{1}{2}} + \nu_{s,n+\frac{1}{2}} \cdot \tau_{n+\frac{1}{2},1}(\bar{\xi}) \widehat{\mathbf{B}}_{s,n+\frac{1}{2}} \otimes \widehat{\mathbf{B}}_{s,n+\frac{1}{2}} \right] \Delta \widehat{\mathbf{d}}_{s,n+\frac{1}{2}} \quad (\text{A.25}) \end{aligned}$$

after an involved calculation found in LAURSEN [18]. The final expressions for the contact stiffness are then given by

$$\mathbf{K}_N^{\text{mat}} = \mathbf{A}_{s=1}^{n_{\text{slave}}} \left\{ \left( \frac{U'(g_{s,n+1}^d) - t_N}{g_{s,n+1}^d - g_{s,n}^d} \right) \widehat{\mathbf{G}}_{s,n+\frac{1}{2}} \otimes \left[ \widehat{\mathbf{G}}_{s,n+\frac{1}{2}} - \frac{c_1}{2} \widehat{\mathbf{D}}_{s,n+\frac{1}{2}} - \frac{c_2}{2} \widehat{\mathbf{T}}_{s,n+\frac{1}{2}} \right] \right\} \quad (\text{A.26})$$

and

$$\begin{aligned} \mathbf{K}_N^{\text{geo}} = \mathbf{A}_{s=1}^{n_{\text{slave}}} \left\{ t_N \left[ \frac{g}{l_s^2} \widehat{\mathbf{N}}_{s,n+\frac{1}{2}} \otimes \widehat{\mathbf{N}}_{s,n+\frac{1}{2}} - \widehat{\mathbf{B}}_{s,n+\frac{1}{2}} \otimes \widehat{\mathbf{D}}_{s,n+\frac{1}{2}} \right. \right. \\ \left. \left. - \widehat{\mathbf{D}}_{s,n+\frac{1}{2}} \otimes \widehat{\mathbf{B}}_{s,n+\frac{1}{2}} + \nu_{s,n+\frac{1}{2}} \cdot \tau_{n+\frac{1}{2},1}(\bar{\xi}) \widehat{\mathbf{B}}_{s,n+\frac{1}{2}} \otimes \widehat{\mathbf{B}}_{s,n+\frac{1}{2}} \right] \right\}, \quad (\text{A.27}) \end{aligned}$$

with the difference quotient in the first term of  $\mathbf{K}_N^{\text{mat}}$  replaced by  $U''(g_{s,n+1}^d)$  if  $g_{s,n+1}^d = g_{s,n}^d$ . We note the non-symmetry of the material part as it occurs with its counterpart for the energy-momentum conserving algorithms considered in this paper for the continuum.

### A.2.2 The tangent contact stiffness

The material part of the tangent contact stiffness is obtained through the linearization of the tangent friction traction  $t_T$ . The expression for  $t_T$  depends on whether we have a stick or a slip condition.

### 1. Stick phase

$$\begin{aligned}\Delta t_T &= \Delta(\kappa_T M_{11} \xi_{n+\theta}^d) \\ &= \kappa_T M_{11} \theta \Delta \xi_{n+1}^d,\end{aligned}\tag{A.28}$$

where

$$\begin{aligned}\Delta \xi_{n+1}^d &= \Delta \left( \hat{H}_{s,n+\frac{1}{2}}^T (\hat{d}_{n+1} - \hat{d}_n) \right) \\ &= \hat{H}_{s,n+\frac{1}{2}} \Delta \hat{d}_{n+1} + \Delta \hat{H}_{s,n+\frac{1}{2}}^T (\hat{d}_{n+1} - \hat{d}_n),\end{aligned}\tag{A.29}$$

and we can recognize that

$$\Delta \hat{H}_{s,n+\frac{1}{2}} = \frac{1}{t_T} K_T^{geo} \Delta \hat{d}_{n+1}.\tag{A.30}$$

Using equation A.29 and A.30, we can rewrite the former as

$$\Delta \xi_{n+1}^d = \left[ \hat{H}_{s,n+\frac{1}{2}} + \frac{1}{t_T} (\hat{d}_{n+1} - \hat{d}_n)^T K_T^{geo} \right] \Delta \hat{d}_{n+1}.\tag{A.31}$$

Finally, using the previous algebraic manipulations, we obtain

$$K_{T,stick}^{mat} = \kappa_T M_{11} \theta \hat{H}_{s,n+\frac{1}{2}} \otimes \left[ \hat{H}_{s,n+\frac{1}{2}} + \frac{1}{t_T} (\hat{d}_{n+1} - \hat{d}_n)^T K_T^{geo} \right]\tag{A.32}$$

### 2. Slip phase

$$\begin{aligned}\Delta t_T &= \Delta \left( \mu t_N \frac{t_T^{trial}}{\|t_T^{trial}\|} \right) \\ &= \mu \Delta(t_N) \frac{t_T^{trial}}{\|t_T^{trial}\|} + \mu t_N \Delta \left( \frac{t_T^{trial}}{\|t_T^{trial}\|} \right),\end{aligned}\tag{A.33}$$

where we know from the previous section that

$$\Delta t_N = - \left( \frac{U' (g_{s,n+1}^d - t_N)}{g_{s,n+1}^d - g_{s,n}^d} \right) \left[ \hat{G}_{s,n+\frac{1}{2}} - \frac{c_1}{2} \hat{D}_{s,n+\frac{1}{2}} - \frac{c_2}{2} \hat{T}_{s,n+\frac{1}{2}} \right] \Delta \hat{d}_{s,n+1},\tag{A.34}$$

with the definition the parameters  $c_1$  and  $c_2$ , and the arrays  $\widehat{T}_{s,t}$  and  $\widehat{D}_{s,t}$ , given above. The expression in parentheses in equation A.34 is replaced by  $U''(g_{s,n+1}^d)$  if  $g_{s,n+1}^d = g_{s,n}^d$ . Also,

$$\Delta \left( \frac{t_T^{trial}}{\|t_T^{trial}\|} \right) = m^{11} \frac{t_T^{trial}}{\|t_T^{trial}\|} \widehat{S}_{s,n+\frac{1}{2}}, \quad (\text{A.35})$$

where we denote by  $\widehat{S}_{s,t}$  the following array:

$$\widehat{S}_{s,n+\frac{1}{2}} = \begin{bmatrix} 0 \\ -N_{,\xi}^{M1}(\bar{\xi})\tau_{n+\frac{1}{2}} \\ -N_{,\xi}^{M2}(\bar{\xi})\tau_{n+\frac{1}{2}} \\ \vdots \end{bmatrix}. \quad (\text{A.36})$$

Thus, we can express the material contribution of the friction force to the overall consistent tangent matrix as follows:

$$\begin{aligned} \mathbf{K}_{T,slip}^{mat} = & -\mu l_s \text{sign}(t_T) \left( \frac{U'(g_{s,n+1}^d) - t_N}{g_{s,n+1}^d - g_{s,n}^d} \right) \widehat{H}_{s,n+\frac{1}{2}} \otimes \left[ \widehat{G}_{s,n+\frac{1}{2}} - \frac{c_1}{2} \widehat{D}_{s,n+\frac{1}{2}} - \frac{c_2}{2} \widehat{T}_{s,n+\frac{1}{2}} \right] \\ & - \mu t_N \frac{\text{sign}(t_T)}{l_s} \widehat{H}_{s,n+\frac{1}{2}} \otimes \underbrace{\left[ \widehat{S}_{s,n+\frac{1}{2}} - \tau_{n+\frac{1}{2}} \cdot \tau_{n+\frac{1}{2},1}(\xi) \widehat{H}_{s,n+\frac{1}{2}} \right]}_{=:\bar{S}_{s,n+\frac{1}{2}}}. \end{aligned} \quad (\text{A.37})$$

For the geometric part of the matrix, we linearize the expression  $\widehat{H}_{s,n+\frac{1}{2}}$ , developed in LAURSEN[18], as follows:

$$\begin{aligned} \Delta \mathbf{H}_{s,n+\frac{1}{2}} = & \frac{1}{m_{11} + g_{s,n+\frac{1}{2}} \nu_{s,n+\frac{1}{2}} \cdot \tau_{n+\frac{1}{2},1}(\bar{\xi})} \left[ 2 \left( \widehat{S}_{s,n+\frac{1}{2}} \otimes \widehat{H}_{s,n+\frac{1}{2}} + \widehat{H}_{s,n+\frac{1}{2}} \otimes \widehat{S}_{s,n+\frac{1}{2}} \right) \right. \\ & - \widehat{G}_{s,n+\frac{1}{2}} \otimes \widehat{D}_{s,n+\frac{1}{2}} - \widehat{D}_{s,n+\frac{1}{2}} \otimes \widehat{G}_{s,n+\frac{1}{2}} - \widehat{T}_{s,n+\frac{1}{2}} \otimes \bar{S}_{s,n+\frac{1}{2}} - \bar{S}_{s,n+\frac{1}{2}} \otimes \widehat{T}_{s,n+\frac{1}{2}} \\ & \left( 3\tau_{s,n+\frac{1}{2}} \cdot \tau_{n+\frac{1}{2},1}(\bar{\xi}) - g_{s,n+\frac{1}{2}} \nu_{s,n+\frac{1}{2}} \cdot \tau_{n+\frac{1}{2},1}(\bar{\xi}) \right) \widehat{H}_{s,n+\frac{1}{2}} \otimes \widehat{H}_{s,n+\frac{1}{2}} \\ & \left. - g_{s,n+\frac{1}{2}} \left( \bar{D}_{s,n+\frac{1}{2}} \otimes \widehat{H}_{s,n+\frac{1}{2}} + \widehat{h}_{s,n+\frac{1}{2}} \otimes \bar{D}_{s,n+\frac{1}{2}} \right) \right] \Delta \widehat{d}_{n+\frac{1}{2}}, \end{aligned} \quad (\text{A.38})$$

where we have defined  $\bar{D}_{s,n+\frac{1}{2}}$  as

$$\bar{D}_{s,n+\frac{1}{2}} = \begin{bmatrix} 0 \\ -N_{,\xi\xi}^{M1}(\bar{\xi})\nu_{n+\frac{1}{2}} \\ -N_{,\xi\xi}^{M2}(\bar{\xi})\nu_{n+\frac{1}{2}} \\ \vdots \end{bmatrix}. \quad (\text{A.39})$$

Thus, the final expression for the tangent matrix is:

$$\begin{aligned} K_T^{geo} = & \frac{t_T}{m_{11} + g_{s,n+\frac{1}{2}}\nu_{s,n+\frac{1}{2}} \cdot \tau_{n+\frac{1}{2},1}(\bar{\xi})} \left[ 2 \left( \hat{S}_{s,n+\frac{1}{2}} \otimes \hat{H}_{s,n+\frac{1}{2}} + \hat{H}_{s,n+\frac{1}{2}} \otimes \hat{S}_{s,n+\frac{1}{2}} \right) \right. \\ & - \hat{G}_{s,n+\frac{1}{2}} \otimes \hat{D}_{s,n+\frac{1}{2}} - \hat{D}_{s,n+\frac{1}{2}} \otimes \hat{G}_{s,n+\frac{1}{2}} - \hat{T}_{s,n+\frac{1}{2}} \otimes \bar{S}_{s,n+\frac{1}{2}} - \bar{S}_{s,n+\frac{1}{2}} \otimes \hat{T}_{s,n+\frac{1}{2}} \\ & \left( 3\tau_{s,n+\frac{1}{2}} \cdot \tau_{n+\frac{1}{2},1}(\bar{\xi}) - g_{s,n+\frac{1}{2}}\nu_{s,n+\frac{1}{2}} \cdot \tau_{n+\frac{1}{2},11}(\bar{\xi}) \right) \hat{H}_{s,n+\frac{1}{2}} \otimes \hat{H}_{s,n+\frac{1}{2}} \\ & \left. - g_{s,n+\frac{1}{2}} \left( \bar{D}_{s,n+\frac{1}{2}} \otimes \hat{H}_{s,n+\frac{1}{2}} + \hat{h}_{s,n+\frac{1}{2}} \otimes \bar{D}_{s,n+\frac{1}{2}} \right) \right] \quad (\text{A.40}) \end{aligned}$$

# Bibliography

- [1] F. Armero and E. Petőcz. A new class of conserving algorithms for dynamic contact problems. *Proc. ECCOMAS 96*, 1996.
- [2] F. Armero and E. Petőcz. Formulation and analysis of conserving algorithms for contact/impact problems. *Comp. Methods in Applied Mech. Engrg.*, 1996 in print.
- [3] F. Armero and E. Petőcz. On the formulation of stable time-stepping algorithms for contact problems. *Proc. COMPLAS V*, 1997.
- [4] F. Armero and E. Petőcz. A new dissipative time-stepping algorithm for frictional contact problems: formulation and analysis. *Comp. Methods in Applied Mech. Engrg.*, 1997 submitted.
- [5] T. Belytschko and M.O. Neal. Contact-impact by the pinball algorithm with penalty and Lagrangian methods. *Int. J. Numer. Methods Engrg.*, 31:547–572, 1991.
- [6] J. Bonet and J. Peraire. An alternating digital tree (ADT) algorithm for 3D geometric searching and intersection problems. *Int. J. Numer. Methods Engrg.*, 31:1–17, 1991.
- [7] N.J. Carpenter, R.L. Taylor, and M.G. Katona. Lagrange constraints for transient finite element surface contact. *Int. J. Numer. Methods Engrg.*, 32:103–128, 1991.
- [8] W.H. Chen and J.T. Yeh. A new finite element technique for dynamic contact problems with friction. *J. de Mécanique théorique et appliquée*, 7-suppl.1:161–175, 1988.
- [9] P. Dworkin. *Efficient Collision Detection for Real-Time Simulated Environments*. Master of science thesis, Massachusetts Institute of Technology, 1994.

- [10] A. Francavilla and O.C. Zienkiewicz. A note on numerical computation of elastic contact problems. *Int. J. Numer. Methods Engrg.*, 9:913–924, 1975.
- [11] L.F. Greengard. *The Rapid evaluation of Potential Fields in Particle Systems*, ACM Distinguished Dissertation 1987. ACM Distinguished Dissertation Series 1987. MIT Press, 1988.
- [12] J.O. Hallquist, G.L. Goudreau, and D.J. Benson. Sliding interfaces with contact-impact in large-scale Lagrangian computations. *Comp. Methods in Applied Mech. Engrg.*, 51:107–137, 1985.
- [13] H.M. Hilber, T.J.R. Hughes, and R.L. Taylor. Improved numerical dissipation for the time integration algorithms in structural dynamics. *Earthquake Eng. and Struct. Dynamics*, 5:283–292, 1977.
- [14] T.J.R. Hughes. *The Finite Element Method, Linear Static and Dynamic Finite Element Analysis*. Prentice Hall, Englewood Cliffs, New Jersey, 1987.
- [15] T.J.R. Hughes, R.L. Taylor, J.L. Sackman, A. Curnier, and W. Kanoknukulchai. A finite element method for a class of contact-impact problems. *Comp. Methods in Applied Mech. Engrg.*, 8:249–276, 1976.
- [16] N. Kikuchi and J.T. Oden. Contact problems of elasticity: a study of variational inequalities and finite element methods. *SIAM*, 1988.
- [17] D.E. Knuth. *The Art of Computer Programming, Fundamental Algorithms, Volume 1*. Addison-Wesley, Massachusetts, 1997.
- [18] T.A. Laursen. *Formulation and Treatment of Frictional Contact Problems using Finite Elements*. PhD thesis, Stanford University, 1992.
- [19] T.A. Laursen and J.C. Simo. On the formulation and numerical treatment of finite deformation frictional contact problems. *Nonlinear Computational Mechanics-State of the Art*, pages 716–736, 1991.

- [20] T.A. Laursen and J.C. Simo. A continuum-based finite element formulation for the implicit solution of multibody, large deformation frictional contact problems. *Int. J. Numer. Methods Engrg.*, 36:3451–3485, 1993.
- [21] K. Lee. Numerical solution for dynamic contact problems satisfying the velocity and acceleration compatibilities on the contact surface. *Computational Mechanics*, 15:189–200, 1994.
- [22] R. Michalowsky and Z. Mroz. Associated and non-associated sliding rules in contact friction problems. *Archives of Mechanics*, 30:259–276, 1978.
- [23] A. Munjiza, D.J.R. Owen, and A.J.L. Crook. Energy and momentum preserving contact algorithm for general 2D and 3D contact problems. *Proc. ECCOMAS 96*, 1996.
- [24] A. Munjiza, D.R.J. Owen, and N. Bićanić. A combined finite-discrete element method in transient dynamics of fracturing solids. *Eng. Comput.*, 12:145–174, 1995.
- [25] J.T. Oden and J.A.C. Martins. Models and computational methods for dynamic friction phenomena. *Comp. Methods in Applied Mech. Engrg.*, 52:527–634, 1985.
- [26] J.T. Oden and E.P. Pires. Algorithms and numerical results for finite element approximations of contact problems with non-classical friction laws. *Computers and Structures*, 19:137–147, 1984.
- [27] H. Parisch. A consistent tangent matrix for three dimensional non-linear contact analysis. *Int. J. Numer. Methods Engrg.*, 28:1803–1812, 1989.
- [28] G.H. Shi. *Block System Modeling by Discontinuous Deformation Analysis*. Computational Mechanics Publications, UK, 1993.
- [29] J.C. Simo and T.A. Laursen. An augmented Lagrangian treatment of contact problems involving friction. *Computers and Structures*, 42:97–116, 1992.
- [30] J.C. Simo and N. Tarnow. The discrete energy-momentum method: Conserving algorithms for nonlinear elastodynamics. *ZAMP*, 43:757–793, 1992.

- [31] J.C. Simo, N. Tarnow, and M. Doblaré. Energy and momentum conserving algorithms for the dynamics of nonlinear rods. *Int. J. Numer. Methods Engrg.*, 38:1431–1474, 1995.
- [32] J.W. Swegle. Search algorithm. Technical Report External Distribution Memo, Solid and Structural Mechanics Department, Sandia National Laboratories, 1993.
- [33] R.L. Taylor and P. Papadopoulos. On the finite element method for dynamic contact/impact problems. *Int. J. Numer. Methods Engrg.*, 36:2123–2140, 1993.
- [34] C. Truesdell and W. Noll. *The nonlinear field theories of mechanics, Handbuch der Physik Bd. III/3*. Springer-Verlag, Berlin, 1965.
- [35] J.R. Williams and R. O'Connor. A linear complexity intersection algorithm for discrete element simulation of arbitrary geometries. *Eng. Comput.*, 12:185–201, 1995.
- [36] P. Wriggers and J.C. Simo. A note on tangent stiffness for fully nonlinear contact problems. *Communications in Applied Numerical Methods*, 1:199–203, 1985.
- [37] P. Wriggers, T. Vu Van, and E. Stein. Finite element formulation of large deformation impact-contact problems with friction. *Computers and Structures*, 37:319–331, 1990.
- [38] Z.-H. Zhong and J. Mackerle. Static contact problems—a review. *Eng. Comput.*, 9:3–37, 1992.